



LAUREA
AMMATTIKORKEAKOULU
Yhdessä enemmän

Kriittiset menestystekijät ohjelmiston tarjoamiseen palveluna

Virtanen, Lassi

2016 Laurea



LAUREA
AMMATTIKORKEAKOULU
Yhdessä enemmän

Laurea-ammattikorkeakoulu

Kriittiset menestystekijät ohjelmiston tarjoamiseen palveluna

Lassi Virtanen
Tietojärjestelmäosaaminen, YTI
Opinnäytetyö
Toukokuu, 2016

Virtanen, Lassi

Kriittiset menestystekijät ohjelmiston tarjoamiseen palveluna

Vuosi	2016	Sivumäärä	47
-------	------	-----------	----

Digitalisaatio on muuttanut ohjelmistojen tuottamista ja tarjoamista merkittävästi viimeisten vuosien aikana. Perinteiset ohjelmistopalveluntarjoajat ovat yhä aktiivisemmin siirtyneet asiakaskohtaisista ratkaisuista tarjoamaan ohjelmistoa palveluna yhdestä, asiakkaiden kesken jaetusta ympäristöstä. Vuonna 2016 digitalisaatio muokkaa maailmaa voimakkaammin kuin koskaan aikaisemmin. Tässä tutkimuksessa digitalisaatio ymmärretään tavoitteena uudistaa vanhoja liiketoimintamalleja ja etsiä uusia digitaalisia palveluita hyödyntämällä.

Yhä useampi ohjelmistotoimittaja miettii ohjelmiston tarjoamista palveluna (Software-as-a-Service, SaaS). Tämän tutkimuksen aikana havaittiin, että vain harvoissa tutkituista yrityksistä tunnistetaan palveluna tarjottavan ohjelmiston menestystekijöitä. Tämän opinnäytetyön tavoitteena on auttaa ohjelmistoa palveluna tarjoavia yrityksiä ymmärtämään ja hyödyntämään palvelukehityksen kriittisiä menestystekijöitä. Tutkimuskysymykseksi tarkentui ”Miten kriittisiä menestystekijöitä voidaan ymmärtää ja hyödyntää ohjelmiston tarjoamiseen palveluna”.

Opinnäytetyö tehtiin tapaustutkimuksena Yinin vuonna 1994 kehittämän mallin mukaan. Opinnäytetyön aineistoa on kerätty vuosien 2013 – 2016 välillä ja opinnäytetyö on kirjoitettu kevään 2016 aikana. Työn taustalla toimii Nunamakerin, Chenin ja Purdinin (1991) malli monimetodologisesta tietojärjestelmäkehityksestä. Työ on tehty vahvasti työelämän kehittämisen näkökulmasta ja tutkimuksen tuloksia on jo alustavasti pystytty hyödyntämään.

Tutkimuksen keskeisinä tuloksina havaittiin neljä kriittistä menestystekijää tarjottaessa ohjelmistoa palveluna. Menestystekijöistä palvelu keskittyy ohjelmistotarjoajan kykyyn vastata asiakkaan odotuksiin. Uuden teknologian hyödyntäminen palvelualussa mahdollistaa ohjelmistotuottamisen palveluna. Kehitysmalli keskittyy kykyyn palvelun jatkuvaan asiakaslähtöiseen kehitykseen. Viimeisenä menestystekijänä mittarit keskittyvät sovelluksen käytettävyyden ja liiketoiminnan mittaamiseen ja näiden tietojen hyödyntämiseen palvelukehityksessä.

Tutkimuksen aikana tärkeimmäksi menestystekijäksi identifioitiin palvelu ja kyky sen asiakaslähtöiseen suunnitteluun. Lisäksi SaaS-palvelun erilaisen liiketoimintamallin ymmärrys sekä kyky sen seuraamiseen olivat olennaisia menestystekijöitä. Näitä tarvitaan liiketoimintapäätöksen tueksi siirryttäessä SaaS-palvelun tarjoamiseen. Näiden lisäksi ohjelmistotoimittajan kyky SaaS-palvelun jatkuvaan kehitykseen sekä palvelun kehityksen ohjaamiseen käytöstä saatavan tiedon perusteella nousivat esille tärkeinä menestystekijöinä. Palvelua tukeva skaalautuva infrastruktuuri on tärkeä näiden mahdollistamiseksi.

Opinnäytetyön lopputuloksena luotiin viitekehys ohjelmistotarjoajan valmiuteen tarjota ohjelmistoa palveluna. Jatkotutkimuksena tullaan seuraavassa vaiheessa tutkimaan SaaS-palveluntarjoajan valmiutta aloittaa palvelun myynti verkossa.

Virtanen, Lassi

Critical success factors for Software-as-a-Service providers

Year	2016	Pages	47
------	------	-------	----

During the last years digitalization has changed radically how companies develop and provide software services to their customers. The goal of digitalization is to renew old business models and find new possibilities when utilizing new technology. Traditional software companies are moving from customer-specific deployments towards providing Software-as-a-Service (SaaS) from one shared environment. In the year 2016 digitalization is changing the world of software providers more than ever before.

At the same time more and more software providers are evaluating the possibility on providing Software-as-a-Service. During this thesis process it was realized that less software providers are aware of the critical success factors when providing software as a service. This thesis aims to help software providers to understand and utilize the critical success factors for Software-as-a-Service. The research question is “How to understand and utilize success factors for providing software as a service”.

This thesis is a case study based on Yin’s (1994) model. The material and sources for this thesis has been collected between 2013 and 2016. The research model basis is from Nunamaker, Chen and Purdin (1991) multimethodological approach to information system research. This thesis has been written in spring 2016. The focus of this thesis is on the development of good practices for software providers’ work life. The results of this study have been utilized towards the end of the thesis process.

Four critical success factors when providing Software-as-a-Service have been identified as the key results of this study. The first critical success factor, service, focuses on the software provider’s ability to create the user experience customers are looking for. The second success factor, technology, focuses on how to provide and operate the service. The third success factor, development model, focuses on how to continually develop the service based on customer feedback and needs. The last success factor, metering, focuses on identifying the critical meters how to develop software and business based on the data from the software.

Based on this study, the most critical success factor when providing Software-as-a-Service is the ability for customer-oriented service design. In addition the ability to understand possible business models in SaaS and the ability to develop business based on the actual business indicators. Also the ability to develop the service based on what customers say and how they use the service and the ability to create and maintain the scalable platform underneath the software were critical success factors when providing Software-as-a-Service.

The result of this thesis is a critical success factor framework for providing Software-as-a-Service. Further research based on this study will focus on defining the Software-as-a-Service provider readiness to sell their services online.

Keywords: Software, Service, SaaS, Software provider, Success factor

Sisällys

1	Johdanto.....	7
1.1	Toimintaympäristö.....	8
1.2	SaaS - Ohjelmisto palveluna.....	9
1.3	Opinnäytetyön tärkeys	11
1.4	Aiemmat tutkimukset.....	12
1.5	Opinnäytetyössä käytetyt lähteet	13
1.6	Opinnäytetyön rakenne	14
2	Tutkimusmetodologia.....	15
2.1	Lähestymistapa.....	16
2.2	Tutkimussuunnitelma.....	17
2.3	Tutkimuskysymys, analyysiyksikkö	18
2.4	Tutkimustulosten analysointi.....	18
2.5	Yhteenveto.....	19
3	Tutkimustulokset	19
3.1	Palvelu	20
3.1.1	Määrittävät tekijät SaaS-palvelun omaksumiseen.....	20
3.1.2	Organisaation valmius SaaS-palvelun käyttöönottoon.....	22
3.2	Palvelun tuottaminen - Soveltuvuus.....	23
3.2.1	Tietojärjestelmän skaalautuvuus.....	23
3.2.2	Tietojärjestelmän resurssien optimointi	24
3.2.3	Tietojärjestelmän saatavuus.....	25
3.2.4	Tietojärjestelmän valinta	26
3.3	Palvelun kehittäminen - Kehitysmalli.....	27
3.3.1	DevOps - Jatkuvan kehityksen malli	28
3.3.2	12 tekijän malli SaaS-palvelun kehittämiseen.....	29
3.4	Palvelun mittaaminen - Mitattavuus	30
3.4.1	Käytettävyyden mittaaminen	31
3.4.2	Liiketoiminnan mittarit	32
3.5	Yhteenveto tutkimustuloksista.....	34
4	Keskustelu	34
4.1	Tutkimuksen tulokset.....	35
4.2	Tutkimusmetodin analysointi.....	38
4.3	Tutkimuksen reliabiliteetti ja validiteetti	38
4.4	Tutkimuksen rajoitteet ja jatkotutkimus	39
5	Lopuksi.....	39
	Lähteet	40

Kuviot	43
Taulukot	44
Liitteet.....	45

1 Johdanto

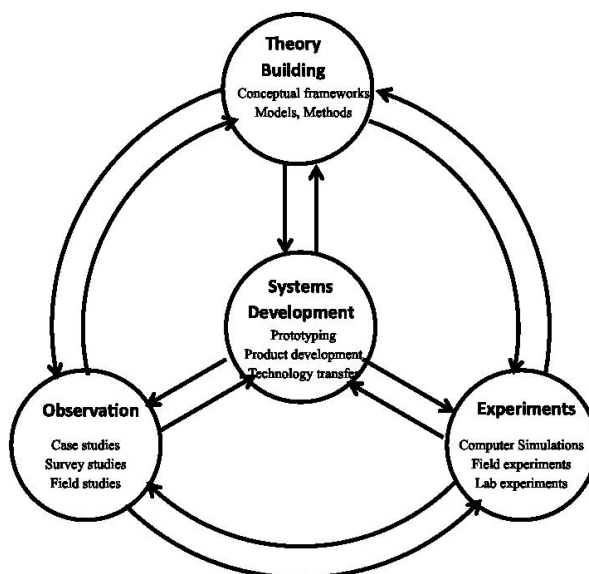
Vuonna 2016 digitalisaatio muokkaa maailmaa voimakkaammin kuin koskaan aikaisemmin. Tässä tutkimuksessa digitalisaatio ymmärretään tapana kyseenalaistaa nykyiset toimintamallit ja auttaa luomaan niistä toimivampia ja joustavampia (Valtiovarainministeriö 2015; Työ- ja Elinkeinoministeriö 2015). Lisäksi digitalisaatiossa hyödynnetään uusia teknologioita, jotta voidaan luoda uusia liiketoimintamalleja ja tulovirtoja (Gartner IT Glossary). Microsoftin toimitusjohtaja Satya Nadella sanoi Microsoftin vuoden 2015 Convergence -tapahtumassa, että jokaisesta yrityksestä tulee ohjelmistoyritys, joka rakentaa sovelluksia, hyödyntää analytiikkaa ja tarjoaa ohjelmistoja palveluna (2015).

Tämän opinnäytetyön tavoitteena on identifioida ja määritellä ohjelmistopalveluntarjoajan menestystekijät sovelluksen tarjoamiseen palveluna (Software-as-a-Service, SaaS). Opinnäytetyön fokus on verkossa tarjottavissa palveluissa. Opinnäytetyö auttaa ohjelmistopalveluntarjoajaa ymmärtämään tarvittavat liiketoiminnalliset ja tekniset menestystekijät muutoksessa kohti ohjelmiston toimittamista palveluna. Mobiilisovellukset on jätetty tämän tutkimuksen ulkopuolelle, sillä niiden myyntilogiikka on vahvasti sidottu suljettuihin markkinapaikkoihin. Koska opinnäytetyö tehdään tietojärjestelmäosaamisen koulutusohjelmassa on myynnilliset ja markkinoinnilliset menestystekijät jätetty työn ulkopuolelle. Näistä saisi itsessään oman jatkotutkimuksen.

Opinnäytetyön aihepiirin tutustuminen ja aineiston keruu on tapahtunut vuosien 2013 ja 2016 välissä. Pääsääntöisesti aineistoa on kerätty tänä aikana Yinin (1994) jaottelun mukaisesti haastatteluissa, suorassa havainnoinnissa ja osallistuvassa havainnoinnissa. Myös tallenteita ja aikaisempia tutkimuksia on kerätty ja analysoitu tänä aikana. Tärkeimmiksi lähteiksi tutkimuksen kannalta osoittautuivat Chenin (2015) tutkimus SaaS-palvelujen hankinnan kyvykkyydestä, Yang (2015) tutkimus organisaatioiden SaaS-valmiudesta sekä Churakova ym. (2010) tutkimus SaaS-liiketoimintamalleista ja ekosysteemeistä. Ammatillinen tausta aihepiirin ympäriltä, hosting- ja pilvipalveluista sekä näiden myynnistä verkossa on vuodesta 2008 alkaen. Työssä yhdistetään useasta eri lähteestä kerättyä aineistoa 3 vuoden ajanjaksolta. Opinnäytetyö koostaa usein erikseen käsiteltyjä osa-alueita, jotka luovat yhdessä toteutettuna riittävät edellytykset ohjelmistoyrityksen muutokseen kohti ohjelmiston tuottamista palveluna.

Opinnäytetyö pohjautuu Nunamakerin, Chenin ja Purdinin (1991) monimenetelmäjäestymiseen tietojärjestelmätutkimuksessa. Mallissa kehitys tapahtuu havainnoinnin, kokeilun ja teorian rakentamisen kautta. Teorian rakentamisessa keskitytään uusiin ideoihin ja konsepteihin. Kokeilussa keskitytään esimerkiksi seuraamaan tietojärjestelmään simulaatioiden kautta. Havainnoinneissa seurataan tapaustutkimuksessa nykyistä tapaa toimia. Lopulta järjestelmän

kehityksessä keskitytään konsepti- ja arkkitehtuurisuunnitteluun, prototyyppiin, kehitykseen ja tuotantoonsiirtoon. (Nunamaker ym. 1991, 91–96.)



Kuvio 1: A Multimethodological Approach to IS Research (Nunamaker 1991, 94)

1.1 Toimintaympäristö

Toimintaympäristöllä on ollut merkittävä vaikutus opinnäytetyön aiheen rajaukseen sekä ajankohtaisuuden validointiin. Vaikka tässä kappaleessa mainitut toimijat ovat pääsääntöisesti työelämän kautta, on opinnäytetyö tehty opiskelijana ja yksityishenkilönä mielenkiinnosta muuttuvaan toimintaympäristöön. Työympäristön kuvaus antaa lukijalle viitteitä tutkimuksen taustalla vaikuttaneisiin tahoihin.

Tutkimuksen kohdeyritys on Pilvi. Pilvi on verkko- ja pilvipalveluiden myyntiin keskittynyt vuonna 2012 perustettu asiantuntijayritys. Pilven tuote on verkkokauppa-alusta pilvipalveluiden myyntiin. Pilven asiakkaita ovat sovellustoimittajat, hosting- ja IT-palveluntarjoajat sekä operaattorit. Yrityksen taustalla on pitkä asiantuntemus verkkokaupasta, hosting- ja pilvipalveluista sekä sovelluskehityksestä. Pilvi tuottaa noin 10 henkilön tiimin voimin oman sovelluksen lisäksi lisäarvoa asiakkaille konsultoimalla sähköisen liiketoiminnan ja pilviautomaation toteutuksissa.

Yrityksessä 2012–2014 tehtyjen aikaisempien tutkimuksien aikana oli havaittu viitteitä suomalaisten ohjelmistoyritysten haasteita siirtymisessä kohti palveluna tarjottavaa ohjelmistoa. Tutkimuksien aikana on tavattu tai haastateltu yli 50 suomalaista ohjelmistoyritystä. Myös vuonna 1993 perustetussa Suomen Ohjelmistoyrittäjät ry:ssä on huomattu samat haasteet.

Ohjelmistoyrittäjien yli 600 jäsenestä vain harva pystyi tuottamaan ja myymään ohjelmistoaan SaaS-mallilla. Pilvi on osallistunut aktiivisesti Ohjelmistoyrittäjien toimintaan ja pyrkinyt vaikuttamaan näin suomalaisten ohjelmistotoimittajien kykyyn tarjota sovelluksia palveluna. Samaan aikaan kuitenkin kansainvälisesti menestyvät ohjelmistot on toimitettu juuri palveluna. Kysymyksenä oli, miten voimme auttaa parhaiten suomalaista ohjelmistoyrittäjää kansainvälistymisessä ja muutoksessa kohti ohjelmiston tarjoamista palveluna. (Ohjelmistoyrittäjät 2015 a.)

Keväällä 2015 Pilvi kutsuttiin yhtenä partnerina suunnittelemaan ja toteuttamaan koulutusohjelmaa suomalaisille ohjelmistoyrityksille edellä mainitun ongelman ratkaisemiseksi. Ohjelma on jaettu neljään aihealueeseen. Aihealueiden jako vastaa hyvin pitkälle tähän opinnäytetyöhön valittuja menestystekijöitä:

1. Planning and implementing SSSaaS (Self-Service Software-as-a-Service) Strategies sekä Positioning and Marketing vastaa menestystekijöistä palvelua;
2. Engineering, Technology, running the services vastaa menestystekijöistä kehitysmallia sekä tuottamista;
3. Ja Customer Acquisition and Metrics vastaa menestystekijöistä palvelun mittaamista (Ohjelmistoyrittäjät 2015 b).

Huomioitavaa on, että tästä opinnäytetyöstä on rajattu ulos myynti ja markkinointi, ja tutkimuksen menestystekijät keskittyvät palvelutuotannon näkökulmaan. Ohjelmistoyrittäjien ohjelma on toteutettu keväällä 2016. Ohjelmaan on osallistunut noin 50 suomalaista ohjelmistoyritystä. Osa tämän opinnäytetyön havainnoista on tullut kyseisen koulutusohjelman kautta. (Ohjelmistoyrittäjät 2015 b.)

Havainnoinnin aikana on huomattu, että itse ohjelmistoyrityksen rooli on muuttumassa vahvemmin pois ratkaisumyynnistä itse sovelluksen kehittämiseen ja tuottamiseen. Konsultoivan ratkaisumyynnin vähentyessä ja myynnin siirtyessä erilaisiin kanaviin (markkinapaikat, jälleenmyyjät, kumppanit) palveluna toimitettavan ohjelmiston arvo kasvaa merkittävästi liiketoiminnan kasvun näkökulmasta. Palvelun käyttöönoton automaatio sekä itsepalvelu tulevat nousemaan ohjelmistotuotannossa merkittävään rooliin.

1.2 SaaS - Ohjelmisto palveluna

SaaS, Software as a Service kääntyy suomeksi parhaiten ohjelmisto palveluna. Ydinajatuksena mallissa on tarjota perinteinen ohjelmisto loppukäyttäjälle Internetin kautta käyttöön perustuvalla laskutuksella palveluna niin, ettei loppukäyttäjän tarvitse asentaa sovellusta omalle päätelaitteelleen. Normaalisti palveluna tarjottavia ohjelmistoja käytetään selaimen tai eril-

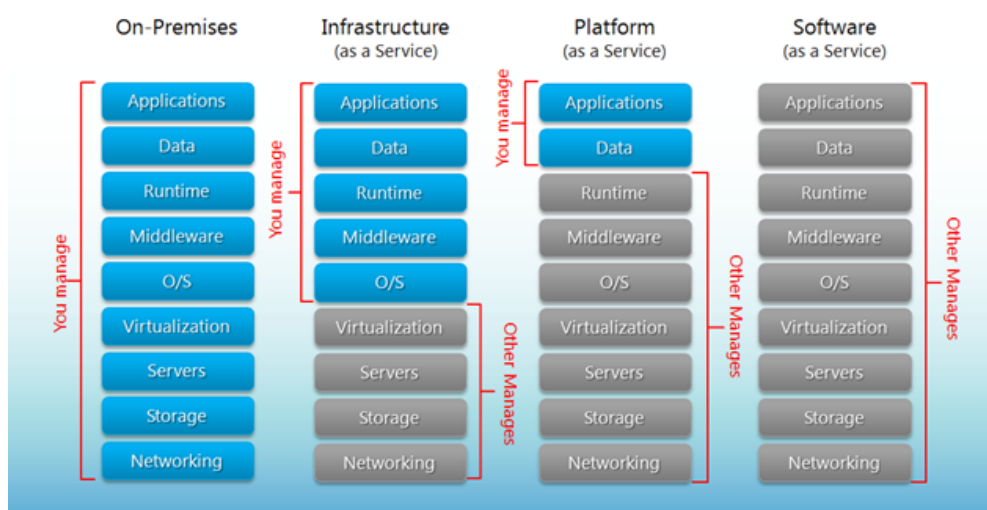
lisen asennettavan päätelaitekohtaisen sovelluksen avulla. Asiakkaan vastuulla SaaS-mallin palvelussa on palvelun käyttöön tarvittavat tunnukset sekä itse palvelun käyttö ja palveluun tallennettava tieto. Yleisin palveluna tarjottavan ohjelmiston laskutusperuste on palveluun luotujen käyttäjätilien määrä. Muita mahdollisuuksia laskutusperusteiksi ovat käyttöön perustuvat laskutukset, useimmiten esimerkiksi tarvittavan tallennustilan määrä.

Yhdysvaltojen kansallinen standardi- ja teknologiainstituutti NIST määrittelee SaaS-mallin seuraavasti:

"The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage or even individual application capabilities, with the possible exception of limited user-specific application configuration settings."
(Mell ym. 2011, 2.)

Käyttötavan ja hallinnan lisäksi SaaS-malliin yhdistetään usein itsepalvelu sekä laajennettavuus. NIST käyttää näistä termejä "On-demand self-service" sekä "Rapid elasticity" (Mell ym. 2011, 2). Näiden avulla SaaS-palvelu on käyttöönotettavissa, maksettavissa ja laajennettavissa verkossa reaaliaikaisesti itsepalveluna ja palvelun käynnistys sekä muutokset palveluun tapahtuvat automaattisesti "without requiring human interaction" (Mell ym. 2011, 2).

Separation of Responsibilities



Kuvio 2: Pilvipalvelun vastuunjako (Remde 2011)

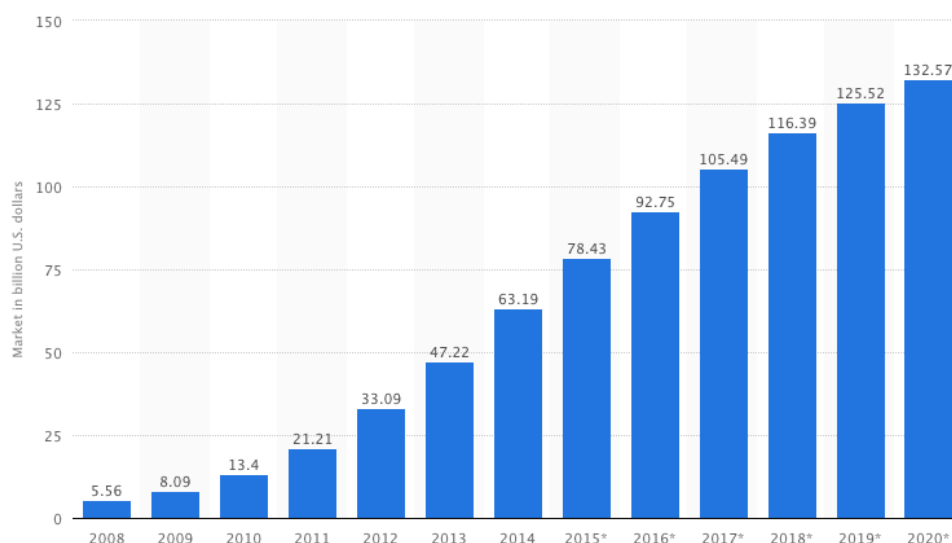
Yllä olevasta Kevin Remden kuvasta nähdään selkeästi eri tyyppisten pilvipalveluiden vastuunjako. Perinteisessä mallissa asiakkaan vastuulla on kaikki ohjelmiston tuottamiseen tarvittava infrastruktuuri palvelinlaitteista lähtien. Ensimmäisessä pilvipalvelutasossa (Infrastructure), palvelinlaitteisto, verkko ja virtualisointi on palveluntarjoajan vastuulla. Itse palvelinohjelmistot, palvelimet ja esimerkiksi skaalautuvuus ovat asiakkaan vastuulla. Toisessa pilvipalvelutasossa (Platform) palveluntarjoaja ylläpitää myös itse palvelimet. Tällöin alusta skaalautuu tarvittaessa käytön mukaan. Viimeisessä tasossa (Software) palveluntarjoaja vastaa kaikesta. Käyttäjän vastuulla on vain itse ohjelmiston käyttäminen.

SaaS-malli on tietävästi ensimmäistä kertaa esitetty yhdysvaltalaisen Software & Information Industry Associationin julkaisussa ”Software as a Service: Strategic Backgrounder” vuonna 2001 (2001). Yleisesti käytetyksi malliksi SaaS-malli on tullut kuitenkin vasta yli 10 vuotta myöhemmin 2010 luvun alussa. SaaS-mallin sovelluksina on ensimmäisenä mainittu asiakkuuksien hallintaan (CRM) tarkoitetut sovellukset. Gartner arvioi vuoden 2007 lopulla CRM-sovellusten kasvavan SaaS-mallilla yli puolet koko markkinaa nopeammin (Gartner 2007). Muina SaaS-mallin esimerkkipalveluina mainitaan esimerkiksi sosiaalisen median palveluista Facebook ja Twitter sekä tiedostopalveluista Google Docs (Sitaram ym. 2012, 173, 185, 193). Opinnäytetyöprosessin aikana tehtyjen havaintojen perusteella näiden palveluiden lisäksi hyvin yleisiä SaaS-mallilla tarjottavia palveluita ovat Googlen Google for Work sekä Microsoftin O365 sähköposti- ja ryhmätyöpalvelut.

1.3 Opinnäytetyön tärkeys

Suurimman viitekehyksen opinnäytetyön tärkeydelle antaa nyt käynnissä oleva aktiivinen keskustelu digitalisaation tuomasta murroksesta. Valtiovarainministeriön mukaan suomalaisilla on EU-maiden paras digiosaaminen. Tämä luo hyvät mahdollisuudet ja kyvyn kyseenalaistaa käytettäviä toimintatapoja ja luomaan ne paremmin asiakkaan ollessa palvelukehityksen ytimessä (2015). Toisesta näkökulmasta katsottuna kansallinen ohjelmistotoimittaja kohtaa digitalisaation myötä kilpailun kansainvälisellä kentällä. Jotta tähän kilpailuun pystytään vastaamaan on ohjelmisto kyettävä toimittamaan verkosta itse käyttöönotettavana palveluna. Gartnerin mukaan juuri SaaS-malli toimii veturina ohjelmistomyyntinä (Gartner 2015).

SaaS-markkinan koosta on lähteestä riippuen erilaisia arvioita. Tutkimuksen aikana on havaittu, että erot arvioissa liittyvät siihen, mikä lasketaan SaaS-mallin liiketoiminnaksi ja mikä ei. Alla oleva Saksalaisen analytiikkayhtiö Statistan julkinen tilasto SaaS-markkinan koosta edustaa numeroissaan ehkä hieman yläkanttiin, mutta antaa realistisen arvion markkinan kasvusta. Lisäksi tutkimusyhtiö IDC:n mukaan SaaS-markkina kasvaa tällä hetkellä lähes 5 kertaa nopeammin kuin perinteinen ohjelmistoliiketoiminta. Statista arvio SaaS-markkinan ylittävän 110 miljardia vuonna 2018 kun IDC arvioi sen tapahtuvan 2019. (IDC 2015; Statista 2016.)



Kuvio 3: Global public SaaS market size 2008 – 2020 (Statista 2016)

Kuten aiemmin kuvastusta toimintaympäristöstä kävi ilmi, iso osa ohjelmistotoimittajista eivät ole kyvykkäitä tarjoamaan sovellusta palveluna. Ymmärrettyään ohjelmiston palveluna tuottamisen menestystekijät tulisi ohjelmistotoimittajan olla merkittävästi kyvykkäämpi tarjoamaan ohjelmistoaan palveluna.

1.4 Aiemmat tutkimukset

Ohjelmiston kehitystä ja ohjelmiston tuottamista palveluna on tutkittu useissa eri tutkimuksissa. Software-as-a-Service -termi on yleistynyt eri tutkimuksissa 2010 vuonna ja sen jälkeen. Aikaisemmista tutkimuksista Gentin yliopistossa julkaistu lopputyö ”Software as a Service - Study and Analysis of SaaS Business Model and Innovation Ecosystems” sekä Elsevierissä julkaistu tutkimuspapere ”Understanding SaaS adoption from the perspective of organizational users: A tripod readiness model” ovat olleet tämän opinnäytetyöprosessin kannalta merkittävimmät tutkimuslähteet. Gentin tutkimuksessa fokus on ollut liiketoimintaympäristön ja liiketoimintamallin ymmärtämisessä. Elsevierissä julkaistussa tutkimuksessa on tutkittu organisatioprosessien ja -kulttuurin vaikutusta kykyyn ostaa ohjelmistoja palveluna. Kummassakaan tutkimuksessa ei ole keskitytty itse ohjelmistotuottajien (tuotannon) menestystekijöihin tarjottaessa ohjelmistoa palveluna. (Churakova 2010; Yang 2015.)

Näiden lisäksi opinnäytetyöprosessin aikana on käyty läpi noin 30 eri tutkimusta, joissa vähintään yhtenä alueena on mainittu ohjelmiston kehitys palveluna (Software-as-a-Service). Tutkimuksissa on käyty läpi mm. palvelun tuottamisen tehokkuuden mittaamista, palvelun tietoturva, liiketoimintaedellytyksiä ja -malleja sekä myyntiä ja markkinointia. Tutkimuksen ai-

kana näitä ei kuitenkaan tunnistettu kriittisiksi menestystekijöiksi, joten näiden tarkempi tarkastelu jätettiin tutkimuksen ulkopuolelle.

Viitekehyksenä opinnäytetyölle olen ollut mukana kolmessa erillisessä tutkimushankkeessa:

Ensimmäisenä Tekes-rahoitteisena tutkimushankkeena elokuusta 2012 joulukuuhun 2013 tutkimme verkkokaupprosesseja pilvipalveluiden myyjän näkökulmasta. Tutkimuksen tavoitteena oli ymmärtää pilvipalveluiden tekninen ja liiketoiminnallinen ympäristö sekä toteuttaa tämän pohjalta verkkomyynti-alusta pilvipalveluiden myyntiin ja hallintaan. Tutkimuksen aikana pystyttiin identifioimaan pilvipalveluiden myynnin ja hallinnan kannalta olennaiset prosessit. Lisäksi saatiin kattava ymmärrys pilvipalveluita ostavien yritysten puolelta pilvipalveluiden hankinnan ja hallinnan haasteista. Tutkimuksen lopputuloksena toteutettiin yrityksen omaan käyttöön edelleen käytössä olevan verkkokauppa-alustan ensimmäinen versio rajoitetuin toiminnallisuuksin.

Toisena Tekes-rahoitteisena tutkimushankkeena elokuusta joulukuuhun vuonna 2014 selvitettiin tarkemmin pilvipalvelumarkkinat sekä kansainvälisesti kiinnostavia kohdemarkkinoita. Tutkimuksen lopputuloksena syntyi suunnitelma sovelluksen kansainvälisen jakelun aloittamiseen. Yhtenä myynnin näkökulmasta kriittisenä menestystekijänä tunnistettiin tällöin tuotannon ja käyttöönoton automaatio, erityisesti juuri silloin kun ohjelmisto tarjotaan palveluna.

Kolmantena tutkimuksena tammikuusta huhtikuuhun 2015 suunniteltiin ja toteutettiin kyselytutkimus ”Pilvipalveluiden hankinta 2015”. Tavoitteena tutkimuksessa oli ymmärtää paremmin suomalaisten pienten ja keskisuurten yritysten (Pk-yritykset) tilaa pilvipalveluiden hankinnan, hallinnan ja haasteiden osalta. Tutkimuksen kohteena oli yli 50 työntekijää työllistävien yritysten IT-päätäjät.

Tämän opinnäytetyön osalta aikaisemmat tutkimushankkeet ovat tuoneet teoria ja ymmärryspohjaa pilvipalveluiden tuottamiseen, käyttöön ja myyntiin liittyen.

1.5 Opinnäytetyössä käytetyt lähteet

Opinnäytetyössä käytetty lähdeaineisto on jaoteltu Yinin (1994) mukaisen tapaustutkimuksen jaottelun mukaisesti (Yin 2009, 101–102; vrt. Kananen 2013, 79):

1. Kirjalliset lähteet (dokumentit, muistiot, raportit, teoriat)
2. Tallenteet (aikaisemmat tutkimukset)
3. Haastattelut (määrittelyt / haastattelut)
4. Suora havainnointi (kohderyhmän havainnointi verkossa)

5. Osallistuva havainnointi (workshopit)
6. Artefaktit (artikkelit ja sähköpostit)

Aineistoa on kerätty vuosien 2013–2016 välisenä aikana. Aineistoa on luokiteltu pääsääntöisesti aineistolähtöisen luokittelun perusteella. Myös teoriapohjaista luokittelua on käytetty. Aineiston luokittelu on opinnäytetyöprosessin aikana elänyt opinnäytetyön rajauksen mukaisesti ymmärryksen ja tiedon määrän kasvaessa. (Yin 2009, 114–124; vrt. Kananen 2013, 104.)

Havainnointi ja haastattelut ovat olleet opinnäytetyöprosessin aikana avainasemassa työn rajaamisessa ja ajankohtaisuuden määrittelyssä. Haastatteluja ja havainnointia on suoritettu tasaisesti koko opinnäytetyöprosessin ajan. Näitä on tehty pääsääntöisesti n. 5 hengen workshopeissa sekä kahdenkeskeisissä haastatteluissa. Yhtä lukuun ottamatta näihin on osallistunut ohjelmistoyritysten liiketoiminnasta, myynnistä tai tuotteen kehityksestä vastaavia asiantuntijoita. Näiden aikana on tehty myös kyselyitä, joilla on pyritty selvittämään ohjelmistoyritysten käytössä olevia teknologioita sekä strategisia tavoitteita ohjelmiston myymiseksi verkossa.

Pääsääntöisesti sähköisiä lähteitä on etsitty Laurean Finna-palvelusta sekä Googlen Scholar-palvelusta. Sähköisissä lähteissä on priorisoitu tieteellisiä julkaisuja ja konferenssijulkaisuja, opinnäytetöitä ja sähköisiä artikkeleita. Lähteitä on etsitty myös Internetistä. Painettuja lähteitä on löydetty pääsääntöisesti Laurean kirjastosta.

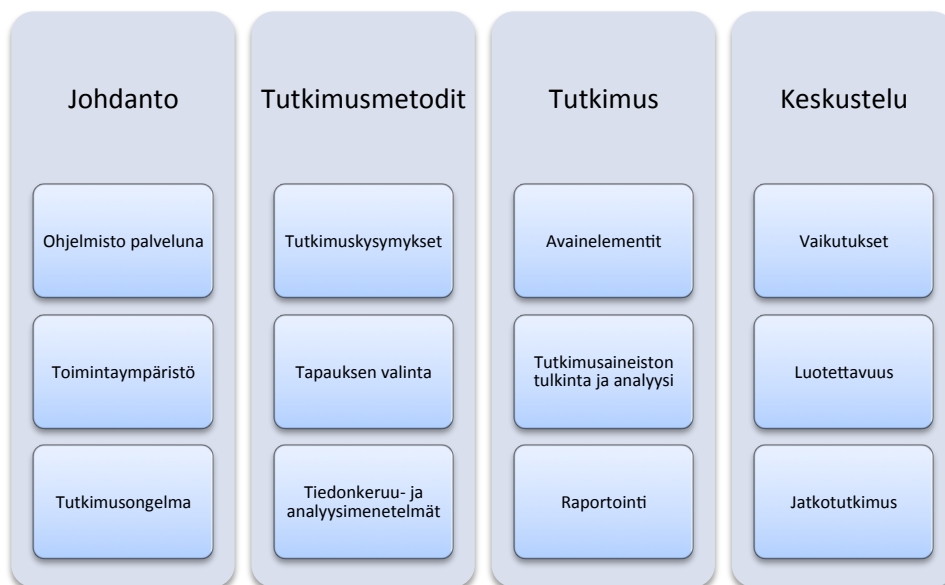
Avainsanoina aineistoa etsittäessä on käytetty mm. Information systems development, Software-as-a-Service, SaaS, cloud service, cloud infrastructure, software performance measurement, automated software delivery, service design, service development. Avainsanoja yhdistämällä ja muokkaamalla on pyritty etsimään teoriaa tukevia lähteitä.

Näiden lisäksi työn lähteinä ovat toimineet aikaisempien tutkimuksien lähteet sekä lopputulokset.

1.6 Opinnäytetyön rakenne

Opinnäytetyön aihepohjana toimii vuosina 2012 – 2015 tehty työ aihealueen parissa. Osa tässä työssä käytetyistä havainnoista ja lähteistä on tältä ajalta. Tutkimusongelman tarkentuessa (markkinamuutoksen tuoma konkreettinen asiakastarve) opinnäytetyö tarkentui alkuvuodesta 2016 koskemaan juuri ohjelmistoyrityksen menestystekijöitä. Samalla pystyttiin tarkentamaan opinnäytetyössä käytettävät ohjelmistoyrityksen menestyksen neljä avaintekijää aiemmissa tutkimuksissa, haastatteluissa, havainnoinneissa ja workshopeissa identifioitujen avaintekijöi-

den pohjalta. Itse tutkimuksessa keskitytään näihin neljään, sovellustoimittajan menestykseen vaikuttavaan avainelementtiin.



Kuvio 4: Työn rakenne mukaillen Yin (1994) tapaustutkimusta (vrt. Yin 2009, 57)

Opinnäytetyön rakenne mukailee Eisenhardtin kahdeksan tutkimusaskelen sekä Yinin tapaus-tutkimuksen vaiheita (Eisenhardt 1989, 533–549; Yin 2009, 57; vrt. Kananen 2013, 59). Aihe-alue ja tästä jalostunut tutkimusaihe on täsmentynyt aikaisempien tutkimuksien sekä tiedonkeruun aikana. Johdannossa käsitellään tutkimusongelmaa ja toimintaympäristöä. Tutkimusmetodeissa käydään läpi tutkimuskysymys sekä tiedonkeruu- ja analyysimenetelmät. Itse tutkimuksessa käydään läpi tutkimuksen tulokset sekä analysoidaan näiden kautta neljää avainelementtiä. Lopuksi käydään läpi tutkimuksen vaikutukset ja luotettavuus sekä jatkotutkimusehdotukset.

Osittain tämän opinnäytetyön pohjalta on aloitettu keväällä 2016 identifioitujen avaintekijöiden soveltaminen konkreettisissa tapauksissa. Kevään aikana löydettyjen avaintekijöiden pohjalta identifioidaan valituissa ohjelmistoyrityksissä valmiutta ohjelmiston tuottamiseen palveluna sekä tehdään konkreettiset toimenpiteet valmiuden saamiseen.

2 Tutkimusmetodologia

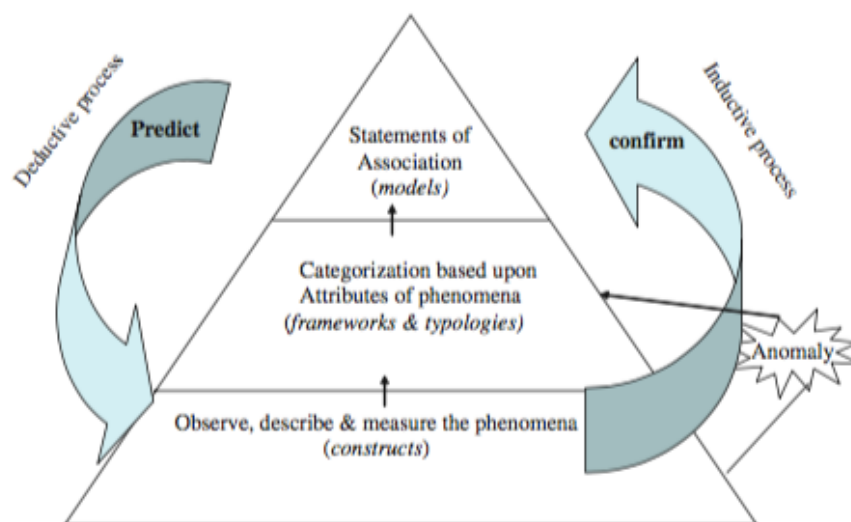
Tässä kappaleessa käsitellään tutkimusmetodin valintaa. Ensimmäisessä kappaleessa käydään läpi tutkimuksen lähestymistapa sekä sitä, miksi tutkimukseen valittiin tapaustutkimus. Seuraavassa kappaleessa käydään läpi tutkimuksen toteutussuunnitelma. Tämän jälkeen avataan tutkimuskysymys sekä analyysiyksiköt. Lopuksi käydään vielä tutkimustulosten analysointia.

2.1 Lähestymistapa

Opinnäytetyön lähestymistapaa on suunniteltu vuodesta 2013 lähtien. Koska opinnäytetyön aihealue on ollut selkeä jo alusta alkaen on lähestymistavan valinta vaikuttanut itse tutkimuskysymyksen muodostumiseen ja tutkimukseen.

Tapaustutkimuksia käsitellään pääasiassa Cunninghamin (1997) ja Eisenhardtin (1989) artikkeleiden sekä Yinin (2009) kirjan perusteella. Tähän opinnäytetyöhön on valittu ensisijaisesti Eisenhardtin kahdeksan tutkimusaskelen sekä Yinin tapaustutkimuksen vaiheiden perusteella mukaillut vaiheet. (Eisenhardt 1989, 533–549; Yin 2009, 57; vrt. Järvinen ym. 2011, 73; vrt. Kananen 2013, 59.)

Lähestymistavaksi valittiin kvalitatiivinen tutkimus, sillä tutkimuskysymyksen tavoitteena on ymmärtää paremmin ohjelmistoyrityksen menestystekijöitä tarjottaessa ohjelmistoa palveluna. Tutkimuksessa edetään induktiolla tutkimusaineistosta kohti teoriaa. Hevnerin ym. mukaan induktiolla pyritään vahvistamaan havainnoinnissa löytyneet mallit. Yinin mukaan tapaustutkimus lähestymistapana on sopiva kun halutaan löytää vastaukset kysymyksiin ”miten” ja ”miksi”. (Eisenhardt 1989, 534–535; Yin 2009, 9, 13; Hevner ym. 2010, 34.)



Kuvio 5: Kuvailevan teorian rakentamisen vaiheet (Hevner 2010, 34)

Tässä tutkimuksessa avainasemassa on ollut aineiston kerääminen sekä havainnointi ja näistä johtopäätösten tekeminen ja teorian vahvistaminen. Riittävällä datamäärällä on pyritty varmistamaan tutkimuksen riittävä saturaatio ja tätä kautta validit loppupäätelmät. Tavoitteena

tutkimuksella oli identifioida induktion kautta menestystekijät sovelluksen toimittamiseksi palveluna.

2.2 Tutkimussuunnitelma

Tutkimussuunnitelman teoriapohjana toimii Eisenhardtin (1989) kahdeksan tutkimusaskelta. Eisenhardtin askeleet on tutkimuksessa jaettu neljään eri osaan: aluksi ja tapausten valinta, välineiden ja työtapojen virittely ja meno kentälle, tietojen analyysi ja hypoteesien hahmotelu sekä suhtautuminen kirjallisuuteen ja prosessin päättäminen. (Eisenhardt 1989, 533; Järvinen ym. 2009, 75–76.)

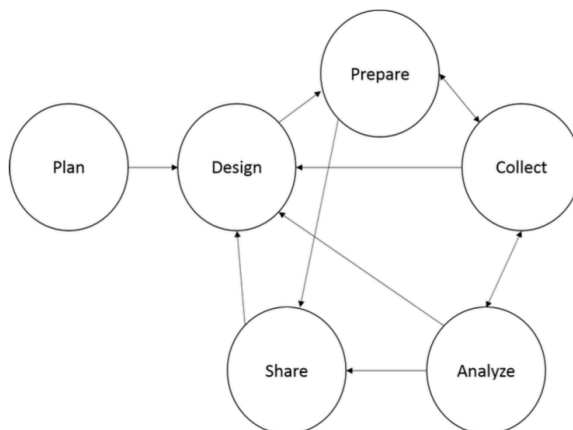
Tutkimuksen ensimmäisessä vaiheessa määriteltiin mahdollisimman tarkka aihio tutkimuskysymykselle. Tutkimuskysymyksellä kohdistetaan tutkimuksen toimenpiteet ja luodaan pohja käsitteiden mittaamiselle. Tässä vaiheessa ei vielä luotu teoriaa tai hypoteeseja. Lisäksi ensimmäisessä vaiheessa valittiin tapaukset, eli mitä tutkitaan. Tavoitteena oli kohdistaa tutkimus konkreettisiin ja hyödyllisiin asioihin. (Eisenhardt 1989, 533, 536–537.)

Tutkimuksen seuraavassa vaiheessa valittiin välineet ja työtavat tutkimusta varten. Tavoitteena oli löytää useita eri datalähteitä ja vahvistaa näin teorian perusteita trianguloimalla lähteitä. Tavoitteena oli myös tutkia eri näkemyksiä ja vahvistaa näin tutkimuksen perustaa. Tämän jälkeen alkoi pitkäkestoinen tietojen keruu ja näiden analysoinnin valmistelu. Tietojen keruun ja analysoinnin aikana myös työtavat kehittyivät. Tietojen luokittelua ja analysointia pystyttiin tehostamaan avainsanojen tarkentuessa tutkimuksen aikana. Aineistoa pystyttiin tehokkaammin kategorisoimaan tutkimuksessa identifioitujen menestystekijöiden taustalle. Tämä auttoi merkittävästi itse opinnäytetyön kirjoitusvaiheessa. (Eisenhardt 1989, 537–539; Yin 2009, 101.)

Kolmannessa vaiheessa osittain päällekkäin toisen vaiheen kanssa suoritettiin tietojen analyysiä kerättyjen datojen perusteella. Tavoitteena oli löytää alustavia teorioita ja jalostaa näitä pidemmälle uusien datalähteiden kautta. Lisäksi muodostettiin hypoteeseja menestystekijöistä, joita tutkittiin myös alustavasti teoriakirjallisuuden kautta. Tavoitteena kolmannen vaiheen aikana oli luoda aihiot menestystekijöille, jotka voidaan viimeisessä vaiheessa validoida. (Eisenhardt 1989, 541–544.)

Viimeinen vaihe jakautuu kahteen osaan: Tutkimustulosten vertaamiseen kirjallisuuteen sekä itse tutkimusprosessin päättämiseen. Viimeisessä vaiheessa tavoitteena on luoda validiteettia ja nostaa tutkimuksen teoreettista tasoa. Prosessin päättämisessä käsiteltiin myös mahdolliset tutkimuksen aikana huomiodut, sen ulkopuolelle jääneet asiat sekä mahdolliset lisätutkimusaiheet. (Eisenhardt 1989, 544–545.)

Tutkimussuunnitelman taustalla toimi Yinin malli iteratiivisesta tapaustutkimuksesta. Tutkimusprosessia kuvaa alla oleva Yinin kuva tapaustutkimuksen toteutuksesta.



Kuvio 6: Doing Case Study Research (Yin 2009, 1)

2.3 Tutkimuskysymys, analyysiyksikkö

Yinin mukaan tapaustutkimus vastaa kysymyksiin ”miten” ja ”miksi”. Kysymyksillä pyritään selittämään syy-seuraussuhteita tai tapahtumaketjuja. Tapaustutkimus soveltuu hyvin tutkit- taessa nykyhetkeä ja jossa tutkijalla on mahdollisimman pieni osuus. Näiden lisäksi tapaustut- kimus sopii vastaamaan kysymykseen ”mitä”. (Yin 2009, 8–13; vrt. Järvinen ym. 2009, 77.)

Opinnäytetyön johdannossa pyritään vastaamaan kysymykseen miksi. Opinnäytetyön tuloksissa pyritään vastaamaan kysymykseen miten. Opinnäytetyön tutkimuskysymys, on miten kriittisiä menestystekijöitä voidaan ymmärtää ja hyödyntää ohjelmiston tarjoamiseen palveluna.

Tutkimuksen analyysiyksiköt ovat:

1. Palvelu (pohjalla on tietojärjestelmä, jonka tietoja toimitetaan palveluna)
2. Palvelun tuottaminen (tietojärjestelmäpalvelun soveltuvuus myytäväksi palveluna)
3. Palvelun kehitys (tietojärjestelmän kehitettävyyys)
4. Palvelun mittaaminen (tietojärjestelmän mitattavuus)

2.4 Tutkimustulosten analysointi

Eisenhardtin mukaan tapaustutkimus on tutkimusstrategia, missä keskitytään ymmärtämään tietyn, valitun tapauksen, nykytilaa. Tapaustutkimuksessa yhdistetään dataa useista eri läh-

teistä. Kerätyn datan analysointi on avain tapaustutkimuksessa luotavaan teoriaan. Kanasen mukaan tapaustutkimuksen ”ensisijaisena tavoitteena on löytää aiheiston avulla ratkaisu tutkimusongelmaan ja vastaukset ongelmasta johdettuihin tutkimuskysymyksiin” (2011, 107). Yin esittää mallissaan viittä tulkintakategoriaa: aikasarja-analyysi, teorian vastaavuus, loogiset mallit, selityksen rakentaminen ja tapausten välinen synteesi. Tämä opinnäytetyö perustuu suurimmilta osin teorian / mallin vastaavuuteen sekä hieman myös loogiseen malliin. Tässä mallissa tutkimustuloksista on koottu looginen kokonaisuus ja tätä on verrattu olemassa oleviin malleihin ja teorioihin. (Eisenhardt 1989, 534; Yin 2009, 136–156; vrt. Kananen 2011, 110–113).

Tutkimustulosten analysoinnin tueksi ja laadun varmistamiseksi käytetty Yinin mukaisesti tietoa useista eri lähteistä. Triangulaation avulla on pyritty varmistamaan tietojen oikeus. Triangulointia hyödynnettiin aineiston, teorian sekä metodologian valinnassa ja todentamisessa. Lisäksi on pyritty ylläpitämään tapaustutkimuksen tietokantaa sekä mahdollisimman aukotonta perusteluketjua. (Miles & Huberman 1994 40–48; Yin 2009, 99–101.)

2.5 Yhteenveto

Opinnäytetyön tapaustutkimukseen on kerätty aineistoa vuosien 2013 – 2016 aikana. Aineistosta on ylläpidetty tapaustutkimuksen tietokantaa ja tietokanta on kehittynyt aineiston keräyksen aikana. Aineistoa on kerätty useista eri lähteistä. Opinnäytetyön aiheen tarkentuessa osa kerätystä aineistosta on todettu myös opinnäytetyön lopullisen aiheen osalta tarpeettomiksi. Opinnäytetyöhön on lähestytty Eisenhardtin ja Yinin mallien mukaisesti.

Lähestymistavaksi valittiin kvalitatiivinen tutkimus, sillä tutkimuskysymyksen tavoitteena on ymmärtää paremmin ohjelmistoyrityksen menestystekijöitä tarjottaessa ohjelmistoa palveluna. Tutkimuksessa edetään induktiolla tutkimusaineistosta kohti teoriaa (vrt. Hevner 2010). Tutkimuskysymykseksi opinnäytetyöprojektin aikana tarkentui ”Miten kriittisiä menestystekijöitä voidaan ymmärtää ja hyödyntää ohjelmiston tarjoamiseen palveluna”.

3 Tutkimustulokset

Tässä kappaleessa käsitellään itse tutkimuksen tuloksia: ohjelmistoyrityksen valmiutta tarjota ohjelmistoa palveluna. Tutkimus on jaettu neljään eri osaan: palveluun, sen tuottamiseen, sen kehittämiseen sekä sen mittaamiseen. Jako on tehty aineistosta nousseen datan perusteella ja pyritty jäsentämään loogisiin kokonaisuuksiin.

Ensimmäisessä kappaleessa käsitellään tutkimustulosten mukaisesti ostajan näkökulmasta palvelun hankintaan vaikuttavia tekijöitä. Opinnäytetyöprosessin aikana haastatteluissa ja

workshopeissa sekä havainnoinneissa juuri palvelumuotoilu havaittiin yhdeksi suurimmista haasteista muutoksessa perinteisestä ratkaisulisenssimyynnistä kohti ohjelmiston tarjoamista palveluna. Kappaleessa käsitellään hankintaa Chenin (2015) tutkimuksen kautta. Tässä tutkitiin SaaS hankinnan ja käytön määrääviä tekijöitä. Toisena tutkimuksena kappaleessa analysoidaan Yang ym. (2014) tutkimusta SaaS-palvelun käyttöönotosta organisaation näkökulmasta kolmikantamallin avulla (Tripod Readiness Model).

Toisessa kappaleessa käsitellään palvelun tietojärjestelmälustaa sovellustoimittajan näkökulmasta. Palvelumuotoilun lisäksi skaalautuvan ja vikasietoisen tietojärjestelmälustan tuottamisessa havaittiin yrityksissä haasteita. Kappaleessa käydään läpi Sitaramin ym., Magedin ym., sekä Abrahamin ym. tutkimusten kautta tietojärjestelmälustan rakentamista skaalautuvuuden, multitenantisuuden sekä saatavuuden näkökulmasta. Lisäksi kappaleen lopussa käydään läpi Tang ym. mallia SaaS-palvelun alustana toimivan tietojärjestelmälustan valintaan. (Abraham ym. 2005; Maged ym. 2007; Sitaram ym. 2012; Tang ym. 2015.)

Viimeisessä kahdessa kappaleessa käsitellään palveluna tarjottavan ohjelmiston kehittämistä sekä mittaamista. Ohjelmistona kehitettävää palvelua on mahdollista toteuttaa useilla eri tavoilla riippuen valituista teknologioista ja palvelutoimittajista. Kehitysmallissa käydään läpi yleisellä tasolla DevOps-kehitysmallia sekä Wigginsin (2012) luoman Twelve-factor app kehitysmallin kautta. Viimeisessä kappaleessa käsitellään SaaS-palvelun kehityksen ohjaukseen liittyvää metriikkaa sekä SaaS-palvelun liiketoimintaohjauksen avainmittareita.

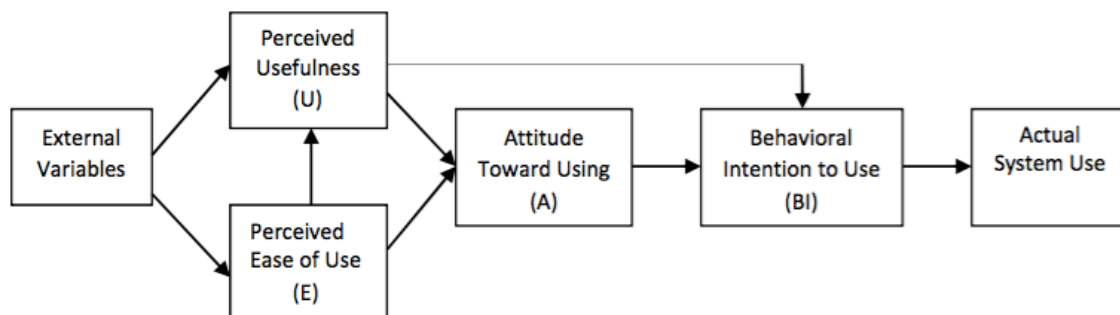
3.1 Palvelu

Lähes poikkeuksetta tutkimuksen aikana havaittiin merkittävimmäksi menestystekijäksi ohjelmiston tarjoamiseen palveluna tuotteistus ja palvelumuotoilu. Perinteisessä ohjelmistomyyntissä toteutettavat ratkaisut kustomoidaan usein asiakaskohtaisesti. Kun ohjelmisto tarjotaan palveluna on itse tarjottava palvelu (tietojärjestelmä) identtinen kaikille asiakkaille. Tällöin korostuu tuotteistuksen ja palvelumuotoilun rooli. Ensimmäiseksi vastataan kysymykseen minkälainen sovellus soveltuu tarjottavaksi palveluna?

3.1.1 Määräävät tekijät SaaS-palvelun omaksumiseen

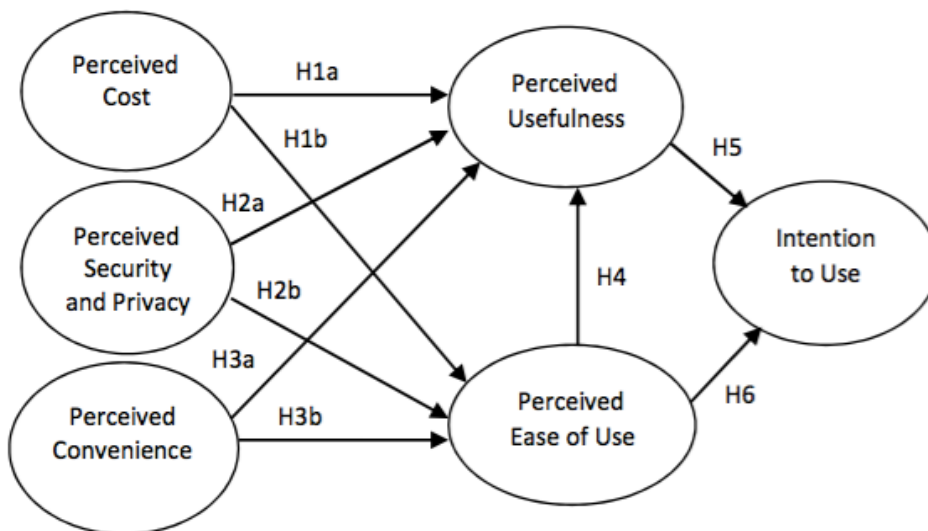
Chen (2015) käyttää tutkimuksensa pohjana Davis ym. (1989) julkaisemaa uuden teknologian hyväksymisen mallia (Technology Acceptance Model). Mallissa portaavat uuden tietojärjestelmän käyttöön ovat koettu helppokäyttöisyys, koettu hyödyllisyys, asenne tietojärjestelmää kohtaan sekä aito aikomus käyttää järjestelmää. Uuden tietojärjestelmän hyväksyminen kiitetty mallissa koettuun hyötyyn (Perceived Usefulness). Uuden tietojärjestelmän täytyy

käyttäjän näkökulmasta helpottaa tehtävää työtä, jotta järjestelmä saadaan aidosti käyttöön. (Davis ym. 1989, 4–5.)



Kuvio 7: Technology Acceptance Model (Davis ym. 1989, 4)

Chenin (2015) tutkimuksessa tutkittiin palveluna ostettavan ohjelmiston ostamisen taustatekijöitä Davis ym. (1989) tutkimuksen pohjalta. Chenin mallissa SaaS-mallilla ostettavan palvelun taustatekijöinä ovat sovelluksen koettu sopivuus tehtävään, koettu tietoturva ja yksityisyys sekä koettu hinta. Näiden pohjalta Chen palaa Davisin (1989) mallin mukaisiin koettuun helppokäyttöisyyteen ja koettuun hyödyllisyyteen. Näiden pohjalta pystytään määrittämään aiotanko sovellusta käyttää.

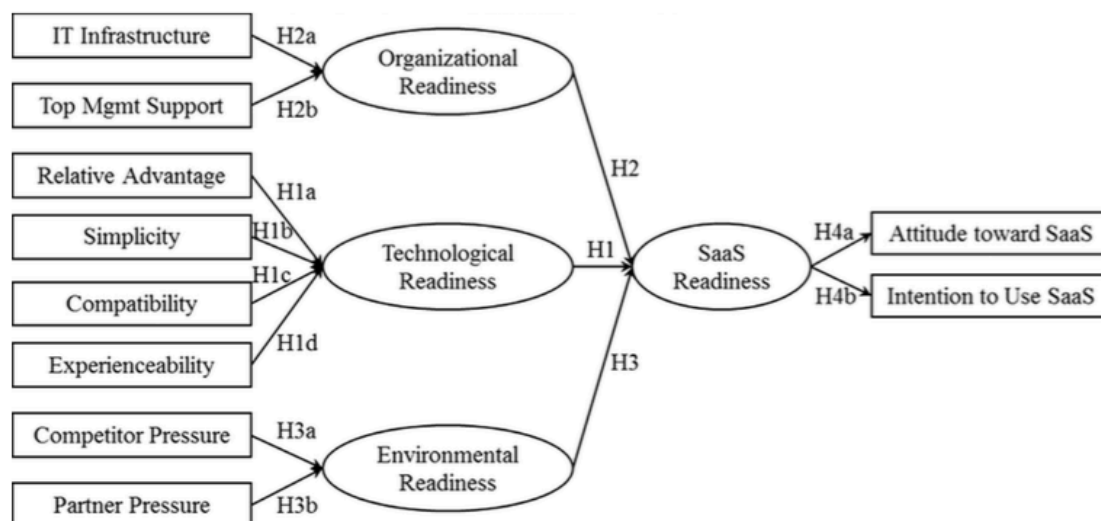


Kuvio 8: Software-as-a-Service Adoption and Intention to Use (Chen 2015, 141)

Chenin tutkimuksen mukaan ainoastaan koetulla hinnalla pystyttiin vaikuttamaan negatiivisesti koettuun helppokäyttöisyyteen ja koettuun hyödyllisyyteen. Palvelun sopivuudella sekä tietoturvalla vaikutettiin käyttöaikomukseen vain positiivisesti. Davisin mallin mukaan myös Chenin mallissa suurin arvo käyttöaikomukseen on koetulla hyödyllisyydellä. (Chen 2015, 146–147.)

3.1.2 Organisaation valmius SaaS-palvelun käyttöönottoon

Chenin keskittyi tutkimuksessaan sovelluksen luomaan kokemukseen ja koettavuuteen. Yang ym. (2014) taas lähestyy tutkimuksessaan organisaation SaaS-valmiutta kolmen eri elementin kautta: Organisaation valmius, teknologinen valmius sekä ympäristön vaikutus valmiuteen.



Kuvio 9: Tripod model of SaaS Readiness (Yang ym. 2015, 256)

Teknologinen valmius viittaa siihen, kuinka valmiita ja halukkaita organisaation käyttäjät ovat ottamaan SaaS-palveluita käyttöön. Perinteiseen ohjelmistoon verrattuna SaaS-mallissa teknologisen valmiuden tulisi olla laajempi, koska ohjelmiston käyttö ylittää totutut fyysiset ja hallinnolliset rajat. Avain teknologiseen valmiuteen on ohjelmiston helppokäyttöisyys sekä ohjelmiston tuomat hyödyt (Relative advantage, Simplicity). Tämän jälkeen merkittävänä tekijänä nostetaan esille yhteensopivuus nykyisten toimintatapojen ja teknologioiden kanssa. Viimeisenä tekijänä teknologiseen valmiuteen nostetaan käyttäjän henkilökohtainen kokemus ohjelmistosta. Tämä nähdään kriittisenä elementtinä, mikä johtaa käyttäjiä suosittelemaan sovelluksen kehitystä ja käyttöä. (Yang ym. 2015, 256–257.)

Organisaation valmiuteen vaikuttaa käytössä oleva IT-infrastruktuuri sekä ylimmän johdon tuki. Tutkimuksen mukaan mitä kehittyneempi IT-infrastruktuuri on, sitä paremmat mahdollisuudet organisaatiolla on kehittää liiketoimintaa SaaS-palveluiden avulla. Ylimmän johdon tuki SaaS-palveluille on erityisen tärkeää pienissä ja keskisuurissa organisaatioissa, jossa hankintapäätös tehdään organisaation hierarkiassa korkealla. Monissa näissä yrityksissä siirtyminen pilvipohjaisiin palveluihin ja SaaS-palveluihin on strateginen päätös. Tutkimuksen mukaan on tärkeää, että ylin johto ymmärtää SaaS-palveluiden hyödyn ja suhtautuu positiivisesti niiden käyttöön. (Yang ym. 2015, 257.)

Kolmantena valmiuteen vaikuttavana elementtinä on yrityksen toimintaympäristö. Toimintaympäristössä suurimpina vaikuttajina ovat kilpailijoiden ja partnereiden luomat paineet. Mitä enemmän kilpailijat hyötyvät liiketoiminnassaan SaaS-palveluiden tuomista eduista, sitä todennäköisimmin organisaation sisälle tulee painetta ottaa SaaS-palveluita käyttöön. Näin kilpailukyky voidaan säilyttää. Partnereiden puolelta paineet tulevat, kun yhteistyössä käytetään yhteisiä IT-järjestelmiä esimerkiksi toimitusketjun hallintaan. Yhteistyö modernien organisaatioiden välillä vaatii usein tiedon vaihtoa partnereiden välillä. (Yang ym. 2015, 258.)

Yang ym. tutkimuksen mukaan kaikki kolme valmiutta (Organisaatio, Teknologia, Ympäristö) ovat välttämättömiä organisaation SaaS-valmiuden selvittämiseksi. Yang ym. mallilla on mahdollista arvioida ja vertailla organisaation valmiuksia SaaS-palvelun käyttöönottoon. Ohjelmistoa palveluna toimittavan yrityksen näkökulmasta tämä tarkoittaa sitä, että on entistä tärkeämpää tuntea ja ymmärtää kohdeasiakkaan liiketoimintaprosesseja, käytössä olevia IT-järjestelmiä sekä organisaatiokulttuuria, jotta voidaan tarjota ohjelmistoa palveluna asiakkaan tarpeisiin. (Yang ym. 2015, 262.)

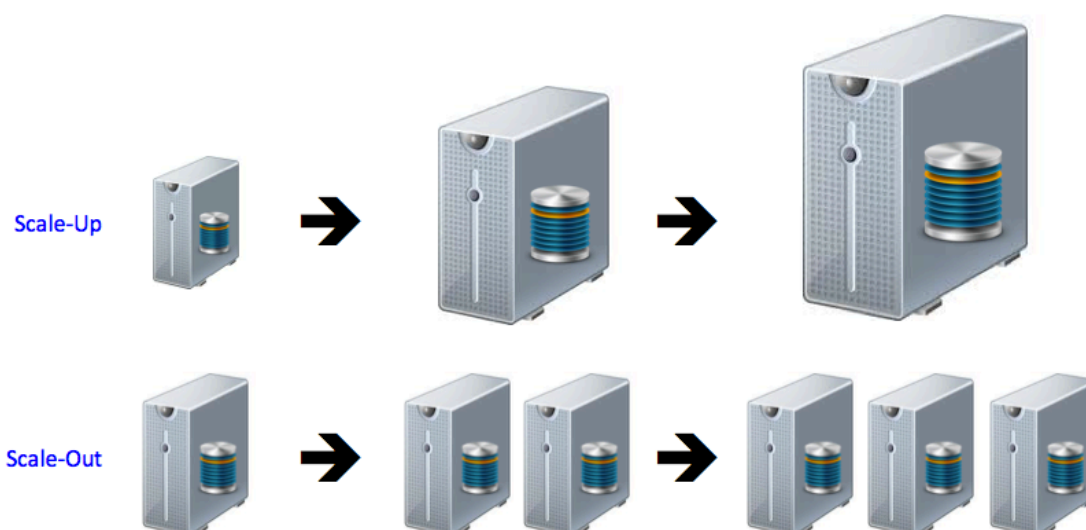
3.2 Palvelun tuottaminen - Soveltuvuus

Tutkimuksen aikana identifioitiin toisena menestystekijänä tietojärjestelmälustan (infrastruktuurin) soveltuvuus SaaS-palvelun alustaksi. Perinteisessä ohjelmistoliiketoiminnassa asiakastoteutukset asennetaan asiakaskohtaiseen ympäristöön, jolloin mahdolliset muutokset infrastruktuuriin ovat hyvin harvoja. SaaS-mallissa asiakastoteutuksia ajetaan yhdestä, tai useammasta jaetusta ympäristöstä. Tällöin uusiksi haasteiksi nousee palvelualustan kyky tarjota palvelua tasaisesti muuttuvalle asiakasmäärälle. Tarjottaessa ohjelmistoa palveluna nousee esille kolme infrastruktuurin avainelementtiä: skaalautuvuus, monen asiakkaan ympäristö samalla infrastruktuurilla (multi-tenancy) sekä palvelun saatavuus. (Sitaram ym. 2012, 255.)

3.2.1 Tietojärjestelmän skaalautuvuus

Palvelualustan skaalautuvuus voidaan toteuttaa pääsääntöisesti kahdella eri tavalla: kapasiteetin lisääminen nykyiseen järjestelmään (scale-up) tai nykyisen järjestelmän laajentaminen (scale-out). Scale-up mallissa nykyisen järjestelmän kapasiteettia kasvatetaan lisäämällä laskentatehoa (CPU), muistia (RAM) tai levytilaa (DISK). Virtualisoiduissa alustoissa kapasiteettia on joissain tapauksissa mahdollista kasvattaa ilman palvelukatkosta tietojärjestelmässä. Useissa tapauksissa kuitenkin palvelu on ajettava alas, jotta kapasiteettia voidaan lisätä järjestelmään. Etuna scale-up mallissa on, ettei itse ajettavalle sovellukselle tarvitse toteuttaa erillistä tukea. (Sitaram ym. 2012, 256.)

Scale-out mallissa tarkoitus on skaalata nykyistä käytössä olevaa kapasiteettia horisontaalisesti lisäämällä uusia tietojärjestelmiä tai solmuja (node) hajautetun tietojärjestelmän käyttöön. Scale-out mallissa tietojärjestelmä on suunniteltu ja rakennettu hyödyntämään tehokkaasti uusia tietojärjestelmiä. Scale-out pohjainen tietojärjestelmä on pohjimmiltaan hajautettu tietojärjestelmä jaetulla levytilalla. (Sitaram ym. 2012, 256–257.)



Kuvio 10: Scale-Up or Scale-Out (Dhandala 2015)

Maged ym. tutki 2007 IBM:n tutkimuksessa scale-up ja scale-out mallin eroja hakupalveluissa. Tutkimuksen lopputuloksena scale-out mallissa oli kiistattomasti parempi hinta ja tehohyöty. Hakupalvelun suuri rinnakkainen työkuorma ja ennakoitavissa oleva prosessori-, verkko- ja levytilakuorma tekee palvelusta soveltuvan hajautettuun tietojärjestelmään. Tutkimuksessa havaittiin, että myös scale-up mallin järjestelmää oli tehokkaampi skaalata scale-out mallilla. Tutkimuksen lopuksi huomioitiin, että hajautetun järjestelmän hallinnointi on hankalampaa kuin keskitetyn järjestelmän. On kuitenkin huomioitava, että tutkimus on tehty lähes 10 vuotta sitten ja kapasiteetin hallinta on kehittynyt tänä aikana merkittävästi. Esimerkkinä yksi pilvipalveluiden hallinnan edelläkävijöistä, OnApp, markkinoi vuoden 2016 pilvitrendeiksi kehittyntä hallintaa sekä automaattista skaalautuvuutta. OnApin mukaan pilven hallintalustat ovat kehittyneet merkittävästi 2010 luvulla. (Maged ym. 2007, 8. OnApp 2016.)

3.2.2 Tietojärjestelmän resurssien optimointi

Toisena SaaS-mallin tietojärjestelmän avaintekijänä tutkimuksen aikana todettiin kyky hyödyntää resursseja tehokkaasti, yhdestä asiakkaiden kesken jaetusta ympäristöstä. Kun monen asiakkaan palvelua tuotetaan samasta jaetusta ympäristöstä, tätä kutsutaan multi-tenanttiseksi ympäristöksi (Multi-tenancy). Asiakaskohtaisissa ratkaisuissa jokaista asiakasta

kohtaan asennetaan asiakaskohtainen tietojärjestelmä asiakaskohtaiseen palvelinympäristöön. Multitenanttisessa tietojärjestelmässä ajetaan yhtä hajautettua tietojärjestelmää, jota käyttää useita asiakkaita samanaikaisesti. Malli kasvattaa tuotannon tehokkuutta merkittävästi, kun ylläpidettävänä ja kehitettävänä on vain yksi versio sovelluksesta. (Sitaram ym. 2012, 284–286.)

Korkeimman tason multitenanttisuuden saavuttaminen voi olla kohtuuttoman kallista kehityksen sekä tietojärjestelmän monimutkaisuuden näkökulmasta. Tutkimuksen aikana todettiin, että palveluna tarjottava ohjelmisto on hyvä suunnitella alusta alkaen tukemaan täyttä multitenanttisuutta, mutta multitenanttisuuden tasoa kasvatetaan tasaisesti palvelun kehittyessä ja asiakasmäärän kasvaessa. Ensimmäisessä vaiheessa toteutetaan asiakaskohtaisia ratkaisuja. Seuraavassa vaiheessa kehitetään yhtä sovellusta, mutta ylläpidetään asiakaskohtaisia ratkaisuja. Kolmannessa vaiheessa ylläpidetään ja kehitetään yhtä sovellusta jaetussa ympäristössä. Viimeisessä vaiheessa varmistetaan tämän ympäristön automaattinen skaalautuvuus tarpeen mukaan. (Sitaram ym. 2012, 285–286.)

Multitenanttisen järjestelmän suurin haaste on resurssien tietoturvallinen jakaminen. Ympäristössä on tärkeä huolehtia autentikoinnista, jolloin käyttäjille varmistetaan pääsy vain hänelle kuuluviin tietoihin. Autentikointi voidaan toteuttaa joko keskitetysti yhden käyttäjäkannan kautta tai hajautetusti usean käyttäjäkannan avulla. Hajautettu autentikointi vaatii luotamusuhteen eri käyttäjäkantojen välillä. SaaS-palveluiden itsepalveluperiaatteen huomioiden keskitetty käyttäjäkanta soveltuu autentikointiin paremmin ja tuntuu olevan laajalti hyväksytty malli. (Sitaram ym. 2012, 287.)

3.2.3 Tietojärjestelmän saatavuus

Kolmantena palveluna tarjottavan sovelluksen alustan kriittisenä elementtinä on palvelun saatavuus. Palvelun saatavuus on taattava toki perinteisessäkin ohjelmiston toimitusmallissa. Tällöin useimmiten asennukset ovat suoraan asiakkaan ympäristöön, jolloin palvelutoimittajan rooli saatavuuden varmistamisessa on pienempi. Kun ohjelmisto tarjotaan palveluna ohjelmistotoimittajan jaetusta ympäristöstä, on vastuu palvelun saatavuudesta ohjelmiston toimittajalla. Mahdollisten katkosten vaikutukset voivat olla myös merkittävämpiä, kun isoja määriä asiakkaita ajetaan samalta hajautetulta palvelualustalta.

Korkean saatavuuden takaaminen on ensisijaisen tärkeää tarjottaessa ohjelmistoa palveluna. Abraham ym. on jakanut korkean käytettävyyden varmistamisen kolmeen elementtiin: Infrastruktuurin saatavuus, sovellusalustan saatavuus (middleware) ja itse palvelun saatavuuteen. Infrastruktuurin saatavuudessa keskitytään itse palvelinlaitteisiin, levyjärjestelmiin ja näitä palvelemaan verkkoinfrastruktuuriin. Sovellusalustan saatavuudessa keskitytään hajautetun

tietojärjestelmän toteutukseen kahdennetuilla tai useaan kertaan varmistetulla ja vikasietoisella sovellusalustalla. Palvelun saatavuudessa keskitytään itse tarjottavan sovelluksen kykyyn tarjota palvelu hajautetusti ja vikasietoisesti, vaikka sovellusalusta tai infrastruktuuri osittain vikaantuisi. (Abraham ym. 2005, 1–4.)

Vaikka tietojärjestelmän alusta (infrastruktuuri, sovellusalusta ja itse palvelu) olisi toteutettu varmistamalla kaikkien elementtien korkea käytettävyys, on silti mahdollista että itse palvelu ei ole saavutettavissa toivotulla tavalla. Vaikka tietojärjestelmä on saatavilla (vastaa kyselyihin) ei se välttämättä toimi niin tehokkaasti kuin käyttäjät olettavat. Lisäksi myös korkean käytettävyyden palveluissa voi olla lyhyitä katkoksia ja tällainen katkos voi osua juuri silloin kun tarve palvelulle on suuri. Viimeisenä esimerkkinä nostetaan palvelun yksittäisten osien käytettävyys. Itse palvelu voi toimia, mutta osia palvelusta tai tietojärjestelmästä voi olla saavuttamattomissa. Tietojärjestelmien muuttuessa entistä monimutkaisemmiksi on tämä muuttumassa entistä kriittisemmäksi ongelmaksi. (Abraham ym. 2005, 1–2.)

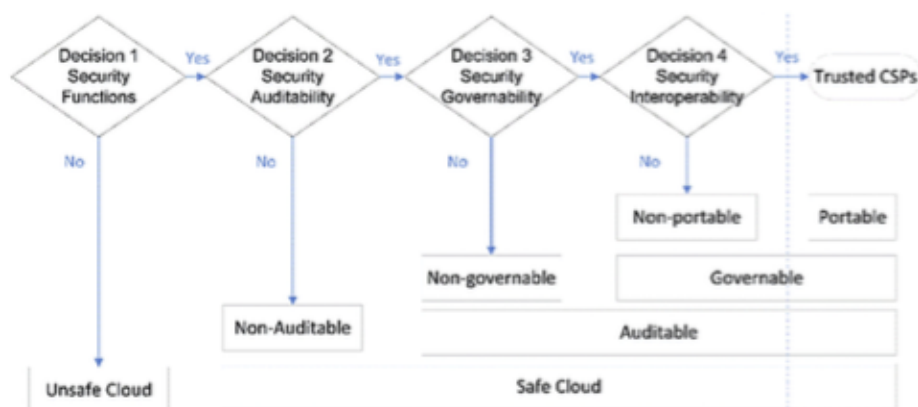
Tietojärjestelmän saatavuuden varmistamiseksi tulee palvelun saatavuutta seurata ja mahdollisiin ongelmatilanteisiin reagoida. Palvelun saatavuutta voidaan seurata pääsääntöisesti kahdella eri tavalla. Ensimmäisessä vaihtoehdossa kysytään säännöllisesti, onko jokin sovittu palvelun osa saatavilla. Tämä malli vaatii useita eri monitorointikohteita, jotta voidaan varmistaa eri tietojärjestelmän osien saatavuus. Mallilla voi kuitenkin olla haastavaa mitata sitä, onko itse palvelu saatavilla kokonaisuudessaan niiltä osin kuin käyttäjä sitä olettaa. Toisessa vaihtoehdossa sovelluksen saatavuutta seurataan kysymällä sovellukselta säännöllisesti jokin ennalta määrätty kysely. Jos palvelu vastaa kyselyyn odotetusti, voidaan todeta, että palvelu on käytettävissä. Saatavuuden seurannassa on myös olennaista, että ongelmien ilmetessä tietojärjestelmä pystyy itsenäisesti joko ohjaamaan liikennettä hajautetun järjestelmän toimiviin instansseihin tai korjaamaan tietojärjestelmää käynnistämällä uusia toimivia instansseja tai palauttamaan sovelluksen toiminta viimeiseen toimivaan pisteeseen. (Sitaram ym. 2012, 298–299.)

3.2.4 Tietojärjestelmän valinta

Hajautetun tietojärjestelmäalustan toteutus itse vaatii merkittävää osaamista sekä taloudellista panostamista itse toteutukseen. Ohjelmistontarjoajalla on mahdollisuus valita useista eri hyvistä palveluntarjoajista itselleen sopivin infrastruktuurin ja sovellusalustan tarjoaja, joka pystyy tarjoamaan tarvittavan skaalautuvuuden, hajautettavuuden sekä saatavuuden.

Tang ym. on 2015 tutkinut luotetun pilvipalvelun tarjoajan valintaa SaaS-palvelun alustaksi. Tutkimuksessa keskitytään palveluntarjoajan valintaan tietoturvan näkökulmasta. Viitekehys soveltuu pienin muutoksin hyvin työkaluksi palveluntarjoajan valintaan teknisen ominaisuuksi-

en näkökulmasta. Pilvipalvelun tarjoajan valinta on jaettu neljään eri vaiheeseen: toiminnallinen soveltuvuus, toiminnallisuuden varmentaminen, hallittavuus sekä yhteensopivuus. Tutkimuksen mukaan systemaattinen lähestyminen luo objektiivisen ja tehokkaan tavan pilvipalvelun tarjoajan valintaan. (Tang ym. 2015, 70–72.)



Kuvio 11: Selecting a trusted cloud service provider for your SaaS (Tang ym. 2015, 71)

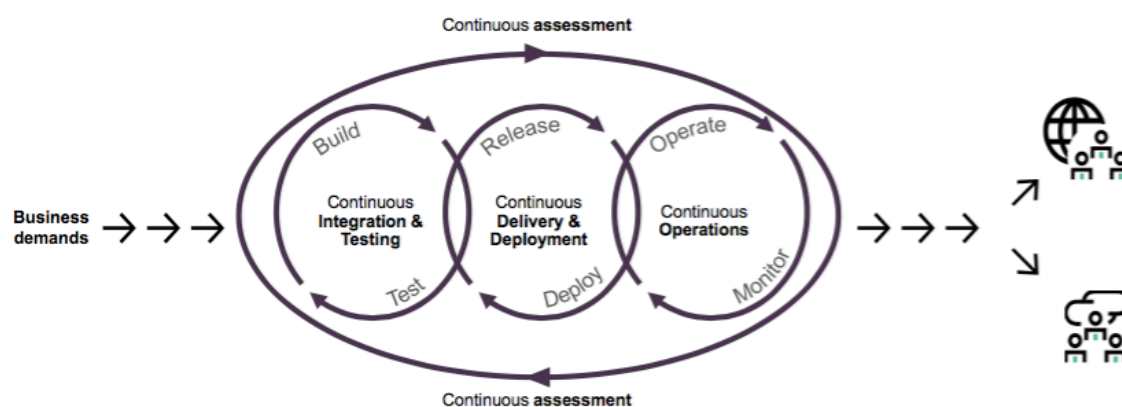
Tietojärjestelmän valinnan pohjalla tulee olla alustava arkkitehtuurisuunnitelma tietojärjestelmästä. Suunnitelman pohjalta voidaan luoda tarvittavat toiminnalliset vaatimukset ja tietoturva-vaatimukset järjestelmän toteuttamiseen. Tämän listan perusteella voidaan listata teknisesti soveltuvat palveluntarjoajat. Kun soveltuvat palveluntarjoajat on listattu, voidaan listaa tarkentaa vertailemalla valittuja palveluntarjoajia määrittelemällä tiettyjä raja-arvoja esimerkiksi toiminnallisuuden todennettavuuden tai sertifiointien perusteella. Tämän jälkeen sopivat palveluntarjoajat voidaan laittaa järjestykseen hallittavuuden ja prosessien näkökulmasta. Miten palveluntarjoajat tarjoavat tukea, miten vikatilanteita hallitaan tai miten hyvin itsepalveluhallinta toimii. Viimeisessä vaiheessa palveluntarjoaja valitaan sen perusteella mikä palveluntarjoajista toimii parhaiten yhteen muiden käytössä olevien tietojärjestelmien kanssa. (Tang ym. 2015, 67–71.)

3.3 Palvelun kehittäminen - Kehitysmalli

Edellisessä kappaleessa käytiin läpi palvelualustan tuottamista. Tässä kappaleessa keskitytään palvelun kehittämiseen. Tutkimuksen aikana havaittiin, että menestyneemmät ohjelmistoa palveluna tarjoavat toimijat ovat omaksuneet tiettyjä malleja ohjelmiston kehitykseen. Toisaalta kuten edellisessä kappaleessa todettiin, vaatii skaalautuvan ja hajautetun palvelualustan toteutus tukea myös itse sovellukselta. Tässä kappaleessa ei keskitytä sovelluskehitykseen teknisessä mielessä. Kappaleessa käydään läpi kaksi eri kehystä sovelluskehityksen ohjaamiseen: jatkuvan kehittämisen malli (DevOps) sekä Wigginsin 2012 julkaisema 12 tekijän malli SaaS-palvelun kehittämiseen.

3.3.1 DevOps - Jatkuvan kehityksen malli

Ohjelmistoa tarjottaessa palveluna yhtenä menestystekijänä tutkimuksen aikana havaittiin kyky ja ajatusmaailma palvelun jatkuvaan kehitykseen. Vuonna 2009 ensimmäisiä kertoja esille noussut DevOps malli antaa yhden hyvän viitekehyksen palvelun jatkuvaan kehittämiseen. DevOps ymmärretään helposti väärin IT-roolina, jossa kehittäjän ja IT-ylläpitäjän roolit ovat yhdistyneet. DevOps on kuitenkin enemmän malli tai viitekehys palvelun kehittämiseen ja julkaisuun. DevOpsilla pyritään luomaan palvelukehitykselle valmius vastata nopeasti muuttuviin liiketoiminnan vaatimuksiin. Gartnerin 2015 julkaiseman DevOps Adoption Surveyn mukaan suurimmat hyödyt DevOps mallissa ovat tuotannon nopeuttaminen sekä nopeampi pääsy markkinoille. Vaikutuksia tutkimuksessa nähtiin myös asiakkaan kokeman arvon nousu, säästöt IT-ympäristössä sekä parantunut riskien hallinta. (Kavis 2014, 163; Sihvonen 2016.)



Kuvio 12: DevOps Prosessi (Sihvonen 2016)

Kaviksen (2014) mukaan DevOps on saanut vaikutteita lean periaatteista (lean manufacturing principles). Yhtenä keskeisimpänä tavoitteena DevOps-viitekehyksessä on maksimoida ohjelmistokehityksen tehokkuus konseptista kehityksen kautta tuotantoon. Tätä varten on identifioitu 6 käytäntöä: infrastruktuurin ja tuotantoonsiirron automaatio, konfiguroitavat toiminnallisuudet (feature flags), mittaaminen, monitorointi ja kyky kokeilla ja epäonnistua nopeassa tahdissa. (Kavis 2014, 165.)

Infrastruktuurin automaatiota käsiteltiin jo edellisessä kappaleessa (Palvelun tuottaminen). Kirjassaan Kavis suosittelee suhtautumaan infrastruktuuriin ja palvelualustaan koodina, sillä palvelualustan luonti ja muutokset palvelualustaan on tehtävissä koodin avulla, jolloin manuaalisia vaiheita palvelualustan muutoksiin ei tarvita. Samalla pystytään varmistamaan hajautetun palvelualustan vaatimuksenmukaisuus ja samankaltaisuus. Itsepalveluna tarjottava infrastruktuuri voi tehostaa kehitystä, mutta huonosti hallittuna aiheuttaa myös merkittäviä haasteita. Automatisoitua käyttöönottoa varten sovelluksen koodi, asiakaskohtaiset konfiguraatiot ja alustan käyttöönottoon tarvittava koodi tulisi olla jaettavissa yhden pisteen kautta.

Automatisoidut käyttöönotot kasvattavat tehtyjen käyttöönottojen määrää ja näin mahdollistavat nopeammat päivityssykliä sovellukseen. (Kavis 2014, 165–167.)

Ohjelmistoa tarjottaessa palveluna jaetusta ympäristöstä ovat konfiguroitavat ominaisuudet (feature flags) yksi merkittävä sovelluskehitystä parantava malli. Mallissa sovelluksen ominaisuudet toteutetaan niin, että jokainen ominaisuus on otettavissa käyttöön, tai pois käytöstä, asiakaskohtaisten konfiguraatioiden kautta. Näin päästään tilanteeseen, missä jokaisella asiakkaalla on käytössä sovelluksesta identtinen koodi ja asiakkaan kanssa sovittu palvelukokonaisuus toteutetaan aktivoimalla konfiguraatiosta vain sovitut osat palvelusta. Toinen merkittävä etu mallissa on, että uusia toiminnallisuuksia voidaan testata tuotannossa rajatulla käyttäjäryhmällä (esimerkiksi vain yrityksen sisäisillä käyttäjillä) ennen laajempaa käyttöönottoa. (Kavis 2014, 167.)

Viimeisenä DevOps viitekehyksen osa-alueena pidetään kehityksen kokeilukulttuuria. Mallissa hyödynnetään sovelluksen käytöstä ja toiminnasta saatavaa tietoa uusien toiminnallisuuksien kehityksessä. Perinteisen toimittajalähtöisen sovelluskehityksen sijaan sovellusta kehitetään käytöstä saatavan tiedon perusteella. Uusien ominaisuuksien toimivuutta voidaan mitata lähes reaaliaikaisesti ja kehittämään näin sovellusta aktiivisesti vastaamaan paremmin käyttäjien odotuksia ja tarpeita. SaaS-palvelun mittaamista käsitellään enemmän seuraavassa kappaleessa. (Kavis 2014, 167–168.)

3.3.2 12 tekijän malli SaaS-palvelun kehittämiseen

Wiggins (2012) on lähestynyt palvelun kehitystä kirjassaan yksittäisten sovelluselementtien kautta ja kerännyt näistä 12 kohdan viitekehyksen. Viitekehys ei ota kantaa sovelluskehityksessä käytettävään teknologiaan tai alustaan. Viitekehys on suunnattu niin sovelluksen kehittäjille, kuin sovelluksen tuotannosta ja ylläpidosta vastaaville ylläpitäjille. Kehyksen tavoitteena on helpottaa uusien kehittäjien aloittamista, sovelluksen hajauttaminen (saatavuuden varmistaminen) eri pilvialustoille, minimoida kehitys- ja tuotantoympäristön eroja mahdollistaen jatkuvan tuotantoonsiirron ja mahdollistaa sovelluksen skaalautuvuus. (Wiggins 2012, 2–3.)

Ensimmäisenä Wiggins nostaa esille Kaviksen (2014) tapaan yhden, versionhallinnassa ylläpidetyn, lähdekoodin merkityksen. Mikäli käytetään useita erillisiä lähdekoodeja tai samasta lähdekoodista ajetaan eri ympäristöissä useita eri sovelluksia ei yhden lähdekoodin malli toteudu. (Wiggins 2012, 4–5.)

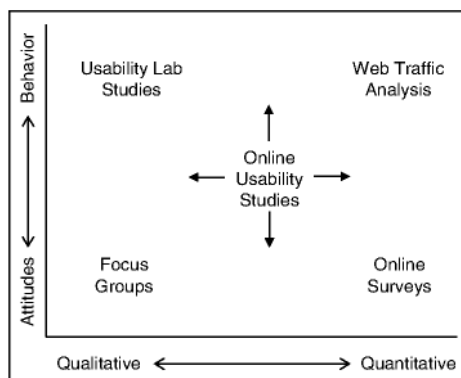
Kaviksen (2014) mukaan myös Wiggins nostaa esille palvelun konfiguroitavuuden. Lisäksi palveluiden taustalla toimivia järjestelmiä tulisi kaikkia käsitellä ulkopuolisina resursseina. Tämä

mahdollistaa muutokset infrastruktuuriin ilman muutoksia sovellukseen. Wigginsin malli rajaa kehitysympäristön, tuotantoonsiirron ja tuotantoympäristön erillisiksi. Ympäristöjen tulisi kuitenkin vastata mahdollisimman lähelle toisiaan. Palvelun skaalautuvuuden suhteen Wiggins listaa ehdottomana tekijänä Maged ym. (2007) mukaisesti scale-out mallin, jolloin nyky-ympäristön kapasiteetin kasvatuksen sijaan luodaan lisää rinnakkaisia prosesseja. (Wiggins 2012, 7–26.)

Wigginsin (2012) Twelve-factor app -malli tukee hyvin aiemmin tässä tutkimuksessa todettuja menestystekijöitä. Malli luo palvelukehitykselle kokonaisen viitekehyksen, jota seuraamalla myös SaaS-palvelun muut menestystekijät esiintyvät palvelussa.

3.4 Palvelun mittaaminen - Mitattavuus

Viimeisenä SaaS-palvelun menestystekijänä tutkimuksen aikana havaittiin kyky ymmärtää ja ohjata palvelukehitystä ja liiketoimintaa aktiivisesti mitattavissa olevan tiedon perusteella. Verkon käytettävyystudkimusmallit voidaan jakaa nelikentällä laadullisiin ja määrällisiin sekä asenteen ja käyttäytymisen erottaviin akseleihin. Tässä kappaleessa keskitytään reaaliaikaisen ja määrällisen käytettävyyden tutkimukseen. (Albert ym. 2010, 5.)



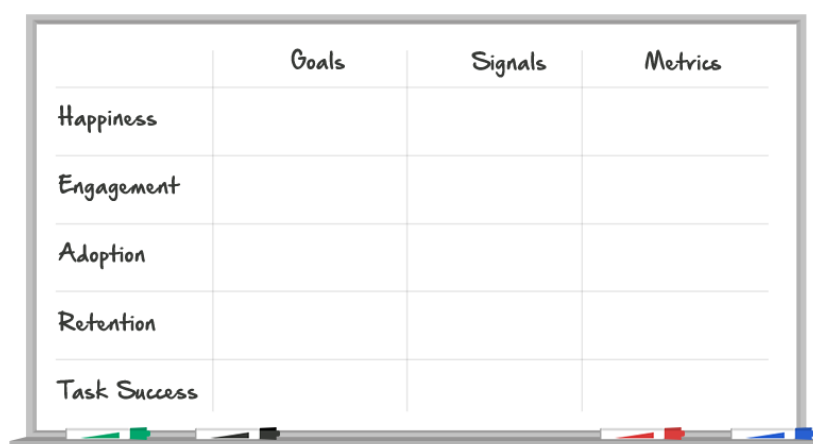
Kuvio 13: Online usability studies (Albert ym. 2010, 5)

Mittareista olennaisimmiksi todettiin kyky mitata asiakkaiden käyttökokemusta, jonka myös Chen (2015) ja Yang ym. (2015) totesivat merkittäväksi SaaS-palvelun käytön kannalta. Tässä kappaleessa käsitellään käyttökokemuksen mittaamista Roddenin (2015) Heart-viitekehyksen kautta. Lisäksi käydään läpi liiketoimintaohjauksen kannalta olennaiset mittarit tarjottaessa ohjelmistoa palveluna.

3.4.1 Käytettävyyden mittaaminen

Jotta palvelua pystytään kehittämään asiakaslähtöisesti, on kyettävä luomaan mittaristo ja seuraamaan asiakkaiden käyttökokemusta palvelussa. Perinteiset web-palvelun mittarit kuten ladattujen sivujen tai uniikkien kävijöiden määrä antavat hyvän pohjan palvelun nykytilalle, mutta ne eivät anna tarvittavaa tietoa palvelun laadusta tai käyttäjän palvelukokemuksesta. Asiakaskokemuksen laadun seurantaan Googlessa on kehitetty HEART-viitekehys ja tämän seuraamista varten Tavoitteet-Signaalit-Metriikka (Goals-Signals-Metrics) -prosessi. (Rodden 2015.)

Artikkelin mukaan käyttökokemuksen mittaaminen onnistuu parhaiten viiden mittarin avulla: käyttäjän tyytyväisyys, sitoutuneisuus ja säilyvyys sekä uusien käyttäjien määrä ja tehtävän onnistuneisuus. Käyttäjien tyytyväisyyttä tai asennetta palvelua kohtaan mitataan usein erillisellä kyselyllä. Kyselyssä voidaan seurata esimerkiksi tyytyväisyyttä tai kuinka helppokäyttöiseksi palvelu on koettu. Sitoutuneisuutta mitataan esimerkiksi sillä kuinka usein yksittäinen käyttäjä käyttää palvelua tai kuinka montaa eri ominaisuutta käyttäjä palvelussa käyttää. Uusien käyttäjien määrää voidaan seurata joko koko palvelun näkökulmasta tai palveluun tuodun uuden ominaisuuden tai toiminnon näkökulmasta. Käyttäjien säilyvyyttä mitataan seuraamalla kuinka moni käyttäjästä käyttää edelleen palvelua. On huomioitava että säilyvyyden mittaaminen on eri asia kuin koko asiakasmäärän mittaaminen. SaaS-palvelun näkökulmasta on olennaista pystyä tunnistamaan ja seuraamaan uusien asiakkaiden sekä poistuvien asiakkaiden määrät erillisinä. Viimeisenä käytettävyyden mittarina mallissa käytetään tehtävän onnistuneisuutta. Tässä voidaan mitata joko onnistuneiden tehtävien määrää ja virheprosenttia tai tehtävään kulunutta aikaa. Tämä mittari on erityisen tärkeä mikäli palvelu on hyvin tehtävuuntautunut. (Rodden 2015.)



	Goals	Signals	Metrics
Happiness			
Engagement			
Adoption			
Retention			
Task Success			

Kuvio 14: HEART - Goals-Signals-Metrics (Rodden 2015)

HEART viitekehys ei yksin riitä tavoitteiden mittaamiseen ja seurantaan. Tätä varten on luotu erillinen Goals-Signals-Metrics mittari. Mittari on rakennettava aina palvelun käyttö ja toiminnallisuudet huomioiden. Ensimmäisenä on tunnistettava tavoitteet, joihin koko kehityksestä vastaava tiimi sitoutuu. HEART viitekehyksen kategoriat antavat hyvän rakenteen tavoitteiden asettamiselle. Yleisimpänä virheenä käytetään vanhoja tavoitteita, jotka eivät auta itse käyttäjäkokemuksen kehittämisessä. Tavoitteiden jälkeen tavoitteet linkataan sovelluksen käytöstä saataviin signaaleihin. Signaali voi olla esimerkiksi tietojärjestelmän tietyn tehtävän parissa käytetty aika (vrt. kuinka monta kertaa kyseinen tehtävä on tehty). Kun tavoitteet on liitetty signaaleihin, voidaan luoda metriikka. Esimerkkinä metriikkaan on käyttäjän keskimäärin tiettyyn tehtävään päivittäin käyttämä aika. (Rodden 2015.)

Käytettävyyksmittariston kehityksessä on ensisijaisen tärkeää luoda mittarit, jotka seuraavat avaintavoitteiden täyttymistä. Mikäli mittaria ei käytetä päätöksiin, on mittari suurella todennäköisyydellä tarpeeton. Näin saadaan luotua mittaristo, jota pystytään hyödyntämään tehokkaasti palvelun kehityksessä. (Rodden 2015.)

3.4.2 Liiketoiminnan mittarit

Viimeisenä ohjelmistoa palveluna tarjoavan yrityksen menestystekijänä tutkimuksessa havaittiin kyky ymmärtää SaaS-palvelun liiketoiminnan avainmetriikkaa. Aiemmin tässä tutkimuksessa käsiteltyjen menestystekijöiden sijaan nämä eivät ole suoraan liitoksissa palvelun kehitykseen, mutta antavat liiketoimintajohdolle kyvyn perustella muutosta kohti palveluna tarjottavaa ohjelmistoa ja johtaa tätä kautta liiketoimintaa (vrt. Yang 2015 ylimmän johdon tuen merkitys). Alla esitetyissä mittareissa keskitytään nimenomaan SaaS-palvelun erityisiin liiketoiminnan mittareihin. Tässä kappaleessa ei oteta kantaa liiketoiminnan yleisiin mittareihin.

SaaS-liiketoiminnan metriikasta puhuttaessa lähes poikkeuksetta tutkimuksen aikana ensimmäisenä mainittiin kuukausittain tai vuosittain toistuvan liikevaihdon määrä (Monthly Recurring Revenue / Annual Recurring Revenue). Mittarissa seurataan jatkuvien tilauksien (subscription) yhteenlaskettua rahallista määrää. Yleisin virhe on laskea jatkuvien tilausten liikevaihtoon mukaan myös kertamaksuja. Jatkuvien tilausten liikevaihtoa voidaan seurata myös tarkemmin erottelemalla jakson aikana tulleet uudet tilaukset, vanhojen asiakkaiden kasvaneet tai pienentyneet tilaukset tai aiemmin poistuneiden asiakkaiden uusintatilaukset. (Skok; Jordan; Lofgren 2012; Chartmogul 2015.)

Toisena merkittävänä mittarina tutkimuksen aikana todettiin asiakaspoistuman seuraaminen (Churn). Mittarissa seurataan kuukausittain peruutettujen jatkuvien tilausten kappalemäärää (Chartmogul 2015; Lofgren 2012). Tämän lisäksi voidaan seurata asiakasmäärän tai toistuvien tilausten liikevaihdon kokonaismuutosta (Jordan ym.; Skok). Lofgren pitää yhtenä raja-arvona

asiakaspoistumassa 10% kuukausittaista poistumaa. Jos poistuma on tätä suurempi, on tuotteessa jotain ”pohjimmiltaan vialla” (Lofgren 2012).

Liiketoiminnan kannattavuuden näkökulmasta havaittiin tutkimuksen aikana tärkeäksi tunnistaa asiakkuuden arvo. Kun seurataan asiakaspoistumaa ja tunnistetaan toistuvan liikevaihdon asiakaskohtainen keskiarvo voidaan laskea asiakkuuden arvo (Customer Lifetime Value). Arvo saadaan jakamalla toistuvan liikevaihdon keskiarvo asiakaspoistuman prosentilla (Chartmogul 2015; Jordan; Skok). Asiakkuuden arvon lisäksi on tärkeä tietää asiakkuuden hankinnan hinta (Customer Acquisition Cost) (Lofgren 2012; Skok). Asiakkuuden hankinta saadaan laskemalla myynnin ja markkinoinnin kulut yhteen ja jakamalla saatu luku aktivoiduilla asiakastileillä (Lofgren 2012; Chartmogul 2015; Skok). Yleisinä arvoina pidetään, että asiakkuuden arvon pitäisi olla vähintään kolminkertainen asiakkuuden hankinnan hintaan ja asiakkuuden hankinta pitäisi kattaa asiakkuuden ensimmäisen vuoden tuloilla (Skok).

Two key guidelines for SaaS startups



Kuvio 15: Two key guidelines for SaaS startups (Skok)

Tutkimuksen aikana viimeisenä avainmittarina huomattiin kyky seurata myynnin konversiota (Funnel). Konversion seuraaminen nousi tutkimuksen aikana selvästi useammin esille niillä yrityksillä, jotka olivat jo pidemmällä valmiudessa tarjota ohjelmistoa palveluna. Jokaisella yrityksellä myynnin konversio on myynnin prosesseista riippuva, mutta ydinajatuksena on seurata asiakkaan siirtymistä markkinoinnin kautta verkkosivuille, kirjautumista palveluun ja lopulta siirtymää maksavaksi asiakkaaksi. Konversion seuraaminen auttaa liiketoiminnan kehityksessä sekä osoittaa myyntiprosessissa kohtia tai vaiheita, jossa asiakaspoistuma on muita kohtia suurempi. (Lofgren 2012; Skok.)



Kuvio 16: SaaS business marketing funnel (Lofgren 2012)

3.5 Yhteenveto tutkimustuloksista.

Alla olevasta taulukosta löytyy kuvattuna tutkimuksen keskeiset tulokset.

Kriittiset menestystekijät ohjelmiston tarjoamiseen palveluna	
Organisaatio	
Kriittinen menestystekijä	Selite
Kyky toimintaympäristön ymmärtämiseen	Kyky havaita ja ymmärtää asiakkailta ja toimittajilta tulevia vaatimuksia ja näiden muutoksia kilpailukyvyn säilyttämiseksi.
Kyky liiketoiminnan ymmärtämiseen	Kyky ymmärtää ohjelmistona tarjottavan sovelluksen liiketoimintamallit sekä kyky hyödyntää näiden avainmittareita liiketoimintapäätöksissä.
Palvelu	
Kriittinen menestystekijä	Selite
Kyky palvelusuunnitteluun	Kyky asiakaslähtöiseen palvelusuunnitteluun ja palvelun yhtenäiseen tuotteistukseen ja hinnoitteluun kaikille asiakkaille
Kyky hyödyntää mitattavaa koettavuutta palvelukehityksessä	Kyky hyödyntää palvelun käytöstä saatavaa mitattavaa tietoa sekä asiakkailta tulevia kehitysehdotuksia palvelun kehityksessä.
Teknologia	
Kriittinen menestystekijä	Selite
Kyky tuottaa palvelualusta	Kyky tuottaa ja ylläpitää hajautettu, korkean käytettävyyden takaava skaalautuva alusta tarjottavalle sovellukselle.
Kyky jatkuvaan kehitykseen	Kyky jatkuvaan yhden palvelun kehitykseen, josta konfiguraatioiden avulla tuotetaan asiakaskohtaista palvelua.

Taulukko 1: Kriittiset menestystekijät ohjelmiston tarjoamiseen palveluna

4 Keskustelu

Tämän tutkimuksen tutkimuskysymys on miten kriittisiä menestystekijöitä voidaan ymmärtää ja hyödyntää ohjelmiston tarjoamiseen palveluna. Tämä kysymys on ohjannut opinnäytetyöprosessia ja työ on rakennettu vastaamaan tähän kysymyksen. Tutkimuksen lopputuloksena kyky ymmärtää ja hyödyntää palvelumuotoilua, palvelun tuottamiseen tarvittavaa infrastruk-

tuuria, kehitysmallia ja mittareita SaaS-palvelun tuottamisessa auttaa ohjelmistopalveluntarjoajaa muuttamaan perinteistä ohjelmistotoimittajan roolia kohti ohjelmiston tarjoamista palveluna.

4.1 Tutkimuksen tulokset

Opinnäytetyönä tehdyn tutkimuksen kautta saatiin kattava käsitys kriittisistä menestystekijäistä tarjottaessa ohjelmistoa palveluna. Ehdottomasti kriittisimpänä menestystekijänä on asiakaslähtöinen palvelusuunnittelu (Chen 2015; Yang 2015). Muina menestystekijöinä tutkimuksen aikana löydettiin kyky kehittää ja tuottaa palvelua sekä kyky mitata palvelun käytettävyyttä sekä palvelun tuomaa liiketoimintaa.

Perinteisessä ohjelmistoliiketoiminnassa palvelu on usein kustomoitu asiakkaan toiveiden mukaisesti. Kun ohjelmistoa tarjotaan palveluna yhdestä jaetusta ympäristöstä, ilman asiakas-kohtaista kustomointia, on palvelun edelleen kyettävä vastaamaan asiakkaan tarpeisiin. Chen (2015) nosti tutkimuksessaan vahvasti esille käyttäjäkokemuksen merkityksen palvelun helppokäyttöisyydestä ja hyödyllisyydestä. Näiden taustalla Chenin tutkimuksessa käyttäjäkokemukseen pystyttiin vaikuttamaan positiivisesti palvelun koetulla sopivuudella ja palvelun koetulla tietoturvalla.

Yang ym. (2015) lähestyi palvelumuotoilua ostavan organisaation SaaS-valmiuden näkökulmasta. Mallissa organisaation SaaS-valmiutta tarkistellaan organisaation, teknologian sekä ympäristön vaikuttavuuden näkökulmasta. Organisaation valmiuteen vaikutti ylimmän johdon tuki. Tämän vaikutus pienissä ja keskisuurissa organisaatioissa nähtiin erittäin suurena. Lisäksi tutkimuksessa havaittiin, että mitä kehittyneempi organisaation IT-ympäristö on, sitä todennäköisimmin organisaatio on valmis ottamaan SaaS-palveluita käyttöön. Teknologinen valmius oli Yang ym. tutkimuksessa lähellä Chen (2015) tutkimusta. Tähän liittyi koettu palvelun helppokäyttöisyys, yhteensopivuus sekä koettavuus. Lisäksi teknologiseen valmiuteen vaikuttaa sovelluksen tuoma hyöty nykytilaan nähden. Viimeisenä Yang ym. käsittelee tutkimuksessaan ympäristön vaikutuksia SaaS-valmiuteen. Ympäristössä erityisesti kilpailuedun säilyttäminen SaaS-palveluita hyödyntämällä nähtiin vaikuttavana tekijänä. Lisäksi yhteistyötahoilta tuleva paine uusien sovelluksien käyttöönottoon nähtiin vaikuttavana tekijänä.

Toisena menestystekijänä tutkimuksessa havaittiin kyky tuottaa palvelua. Perinteisessä ohjelmistoliiketoiminnassa asiakastoteutukset asennetaan asiakaskohtaiseen ympäristöön. Palveluna tarjottavassa ohjelmistossa asiakkuuksia ajetaan yhdestä jaetusta tietojärjestelmästä ja sen tuotantovastuu on palvelun toimittajalla. Infrastruktuurin soveltuvuuteen SaaS-palvelun alustaksi havaittiin vaikuttaviksi tekijöiksi palvelualustan skaalautuvuus ja multitenantisuus sekä kyky varmistaa palvelun saatavuus ja valita sopiva palveluntarjoaja.

Tietojärjestelmää on mahdollista skaalata joko lisäämällä kapasiteettia nykyiseen järjestelmään (scale-up) tai lisäämällä rinnakkaisia prosesseja (scale-out). Maged ym. (2007) tutki IBM:n tutkimuksessa palvelun skaalautuvuutta. Tutkimuksessa havaittiin kiistaton hinta ja tehohyöty scale-out-mallissa. Myös Wiggins (2012) suositteli scale-out mallia ohjelmistoa tarjottaessa palveluna. Magedin ym. tutkimuksessa korostettiin kuitenkin, että hajautetun järjestelmän ratkaisujen hallinta on haastavampaa. Tutkimus on tehty kuitenkin lähes 10 vuotta sitten ja nyt tehdyn tutkimuksen aikana havaittiin että hajautetun kapasiteetin hallinta on kehittynyt vuodesta 2007 merkittävästi.

Palveluna tarjottavan tietojärjestelmän toiseksi menestystekijäksi havaittiin kyky tarjota palvelu yhdestä tai useammasta asiakkaiden kesken jaetusta ympäristöstä (multitenanttisuus). Tavoitteena on tarjota ja ylläpitää vain yhtä sovellusta ja varmistaa skaalaamisella tietojärjestelmän resurssien riittävyys. Tutkimuksen aikana havaittiin, että täydellisen multitenanttisuuden saavuttaminen on sekä osaamisen että tietojärjestelmän kustannusten kannalta haastavaa. Tärkeäksi havaittiin kyky nostaa multitenanttisuuden tasoa asiakasmäärän ja palvelukäytön kasvaessa. Suurimpana haasteena multitenanttisessa järjestelmässä pidettiin tietoturvan toteuttamista ja turvallisen autentikoinnin toteuttamista pidettiin hyvin tärkeänä. (Sitaram ym. 2012.)

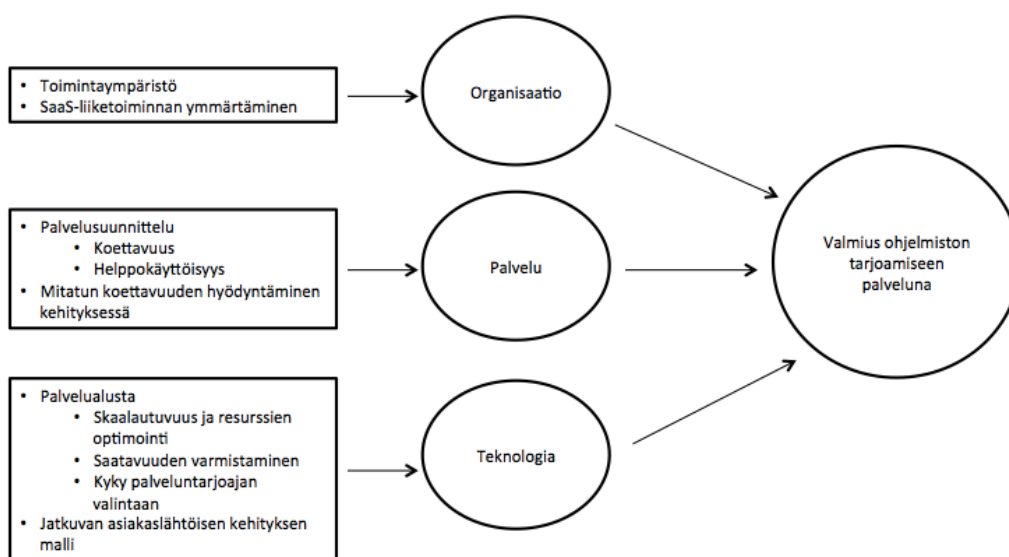
Kolmantena tietojärjestelmän menestystekijänä havaittiin kyky varmistaa palvelun saatavuus. Abraham ym. (2005) tutkimuksessa palvelun saatavuus jaetaan kolmeen osaan: infrastruktuurin saatavuus, sovellusluston saatavuus ja itse palvelun saatavuus. Infrastruktuurin saatavuudessa keskitytään itse palvelinlaitteiston, levyjärjestelmän ja verkon saatavuuteen. Sovellusluston saatavuudessa keskitytään palvelun alla toimivien sovellusten saatavuuteen ja itse palvelun saatavuudessa keskitytään palvelun saatavuuteen ja käytettävyyteen. Tietojärjestelmän saatavuuden varmistamista pidettiin jatkuvasti vaikeutuvana tehtävänä tietojärjestelmien monimutkaistuessa. Sitaram ym. (2012) mukaan saatavuuden varmistamisessa voidaan joko seurata tietojärjestelmän yksittäisten osien toimintaa tai seurata tietojärjestelmän kykyä suorittaa jonkin ennalta määrätty tehtävä.

Skaalautuvuus, multitenanttisuus ja saatavuus ovat keskeisiä elementtejä tietojärjestelmäalustaa valittaessa. Tang ym. (2015) tutki luotetun pilvipalvelun tarjoajan valintaa SaaS-palvelun alustaksi. Tutkimuksessa käytettiin neliportaista valintametodia: Soveltuvuus, luotettavuus, hallittavuus ja yhteensopivuus. Vaikka Tang ym. on ensisijaisesti luonut viitekehyksen tietoturvan näkökulmasta, voidaan sitä käyttää hyvin myös palveluntarjoajan valintaan teknisestä näkökulmasta.

Kolmantena menestystekijänä tutkimuksen aikana havaittiin kyky kehittää palvelua hyväksi havaittujen mallien mukaisesti. Ensimmäisenä kehityksen kannalta olennaisena menestystekijänä havaittiin kyky vastata liiketoiminnan nopeisiin muutoksiin jatkuvan kehityksen mallilla. Tässä työssä mallina käytettiin laajasti omaksuttua DevOps-viitekehystä, mikä on Kavis (2014) mukaan kulminoitunut kuuteen käytäntöön: infrastruktuurin ja tuotantoonsiirron automaatio, konfiguroitavat toiminnallisuudet, mittaaminen ja monitorointi sekä kyky kokeilla ja epäonnistua nopeassa tahdissa. DevOps kehitysmalli sekä Wigginsin (2012) luoma Twelve-factor app -malli tukevat hyvin aiemmin tutkimuksessa havaittuja menestystekijöitä palvelusuunnittelusta ja palvelun tuottamisesta.

Viimeisenä menestystekijänä tutkimuksessa havaittiin kyky mitata käytettävyyttä ja liiketoimintaa, ja johtaa kehitystä mitatun tiedon perusteella. Tutkimuksessa mitattavuus jaettiin käytettävyyden mittaamiseen sekä liiketoiminnan avainmittareihin. Palvelun käytettävyydessä avainmittareiksi todettiin käyttäjän tyytyväisyys ja sitoutuneisuus, muutokset palvelun käyttäjämäärissä sekä onnistuneet tehtävät palvelussa. Rodden (2015) artikkelissa käytettävyyden seuraamista ja tällä tiedolla tuotekehityksen johtamista pidettiin tärkeänä.

Viimeisenä menestystekijänä käytiin läpi SaaS-liiketoiminnan avainmittareita. Yang ym. (2015) piti tutkimuksessaan yhtenä SaaS-käyttöönoton merkittävänä mittarina ylimmän johdon tukea. Myös tämän opinnäytetyön aikana havaittiin kriittiseksi menestystekijäksi kyky ymmärtää ja käyttää SaaS-liiketoiminnalle olennaisia mittareita. Avainmittareina tutkimuksessa tunnistettiin jatkuvan liikevaihdon seuraaminen, asiakaspoistuman seuraaminen sekä liiketoiminnan kannattavuuden seuraaminen. (Skok; Jordan; Lofgren 2012; Chartmogul 2015.)



Kuvio 17: Ohjelmistoyrityksen SaaS-valmius

Tutkimuksen tulokset voidaan kiteyttää yllä olevalla SaaS-valmiutta kuvaavalla viitekehyksellä. Viitekehvyssä kuvataan kriittisimmät menestystekijät organisaation, palvelun ja teknologian näkökulmista. Malli viitekehvykseen on otettu Yang ym. (2015) SaaS-valmiuden kolmikantamallista.

4.2 Tutkimusmetodin analysointi

Tapaustutkimuksen ensisijaisena tavoitteena on löytää ratkaisu tutkimusongelmaan ja tästä tehtyihin tutkimuskysymyksiin. Tutkimuksessa on etsitty Yinin mallin mukaan tyypillistä tapaa toimia tai toiminnan samanlaisuutta. Tutkimusaineistosta on löydetty menestystekijöitä, jotka yhdistävät ohjelmistoa palveluna tarjoavia palvelutuottajia. Löydettyjä menestystekijöitä on pyritty varmentamaan havaintojen lisäksi teoriapohjan kautta. Tutkimusta on tehty induktion kautta. Tutkimustuloksia on siis lähestytty aineiston ja sen luokittelun kautta. (Yin 2009, 136–159; vrt. Kananen 2013, 107, 109–110.)

Opinnäytetyöprosessi ja tutkimus on kestänyt noin kolme vuotta. Tänä aikana on pystytty havainnoimaan kirjallisten lähteiden, tallenteiden, haastattelujen, havainnoinnin ja artefaktien kautta tutkimuskysymyksen menestystekijöitä. Tutkimukseen on kerätty aineistoa laajasti ja tutkimusta on tehty läheisessä yhteistyössä työelämän kanssa. Opinnäytetyön viimeistelyn aikana on pystytty todentamaan löydettyjen menestystekijöiden vaikutusta ohjelmistoyrityksen kykyyn tarjota ohjelmistoa palveluna.

Tapaustutkimuksen valintaa tutkimusmetodiksi tuki vahvasti opintojen aikana käyty tapaus-tutkimuksen kurssi. Koska tutkimuksessa etsittiin vastausta kysymykseen miksi, oli luontevaa valita metodiksi tapaustutkimus. Tapaustutkimus voidaan kategorisoida viiden eri kategorian kautta: aikasarja-analyysi, teorian tai mallin vastaavuus, loogiset mallit, selityksen rakentaminen ja tapausten välinen synteesi (Yin 2009, 136–160). Tämän tutkimuksen aikana on pyritty ymmärtämään ja hyödyntämään mallia, kriittisiä menestystekijöitä, ohjelmiston kehittämiseen palveluna.

4.3 Tutkimuksen reliabiliteetti ja validiteetti

Yin (2009) mukaan tapaustutkimuksen tulosten ja johtopäätösten pitäisi olla oikeita ja luotettavia (”logical set of statements”). Yin nostaa kirjassaan luotettavuuden arviointiin laadullisessa tutkimuksessa rakenteen oikeellisuus, sisäinen ja ulkoinen pätevyys sekä luotettavuus. Tässä työssä reliabiliteetti ja validiteetti on pyritty varmistamaan seuraamalla Yin (2009) kuuden lähteen mallia (six sources of evidence) Ristiriidattomuuden varmistamiseksi kerätystä

aineistosta on pyritty identifioimaan samat johtopäätökset usean eri toisistaan riippumattoman lähteen kautta. Saturaatiolla on pyritty varmistamaan, että sama johtopäätös toistuu riittävän usein. Arvioitavuutta on pyritty varmistamaan kattavalla lähteiden käytöllä ja lähdeviitteiden merkinnällä. (Yin 2009, 40–45, 101–103).

4.4 Tutkimuksen rajoitteet ja jatkotutkimus

Opinnäytetyö on tehty tietojärjestelmäosaamisen koulutusohjelmassa. Työstä on tarkoituksellisesti rajattu ulos liiketoiminnalliset menestystekijät (myynti ja markkinointi) sekä tietoturva. Molemmat näistä ovat nousseet tutkimuksen aikana esille kriittisten menestystekijöiden taustalla. Molemmat ovat kuitenkin aiheina niin laajoja, että niitä ei pystytty tämän tutkimuksen puitteissa tutkimaan

Tämän tutkimuksen pohjalta suosittelen keskittymään jatkotutkimuksessa palvelusuunnitteluun - palvelumuotoilu, tuotteistus, hinnoittelu - sekä tietoturvallisen tietojärjestelmäalustan valintaan ja käyttöönottoon. Tutkimuskysymyksiä voisi esimerkiksi olla, miten voidaan ymmärtää ja hyödyntää palvelun käytettävyydestä saatavaa metriikkaa palvelun kehityksessä tai miten voidaan ymmärtää ja hyödyntää palvelun tietoturva vaatimuksia kansainvälisen SaaS-palvelun toteutuksessa. Tämä auttaisi ohjelmistoyrityksiä ymmärtämään kattavammin myynnin ja tuotannon menestystekijöitä tarjottaessa ohjelmistoa palveluna.

5 Lopuksi

Tutkimuksen toteutuksessa on ollut kannustavaa havaita aidosti tutkimusongelman ajankohtaisuus. Tilastot eri lähteistä osoittavat vahvaa markkinakasvua. Toisaalta tutkimuksen aikana haastatteluiden lisäksi eri lähteistä tunnistettu aito ajankohtaisuus ja tarve ymmärtää paremmin SaaS-mallin menestystekijöitä on auttanut opinnäytetyön toteutusta.

Opinnäytetyöprosessi on ollut pitkä ja äärimmäisen opettava. Hieman yllättävänä tekijänä on ollut erityisen hienoa huomata viime kuukausien kirjoitusprosessin aikana eri teoreettisten viitekehysten tuki työelämän tehtävien tukena. On ollut myös opettavaa vaihtelua tehdä kattavampaa tutkimusta. Suurimpana haasteena opinnäytetyöprosessissa oli tutkimuksen kohdennus ja lopullinen tutkimuskysymyksen asettelu.

Opinnäytetyöprosessi ei olisi tässä ajassa valmistunut ilman Laurean yliopettaja Rauno Pirisen tukea, opintovapaan mahdollistamaa joustoa kirjoituksen viime metreillä sekä perheen ja muiden ystävien joustoa vapaa-ajan kuluessa opinnäytetyön parissa.

Lähteet

Painetut lähteet:

Albert, B. Tullis, T. Tedesco, D. 2010. Beyond the Usability Lab: Conducting Large-scale Online User Experience Studies. Morgan Kaufmann Publishers

Chaffey, D. 2011. E-Business & E-commerce Management - Strategy, Implementation and Practice. 5. Painos. Prentice Hall

Kananen, J. 2013. Case-tutkimus opinnäytetyönä. JAMK

Kavis, M. 2014. Architecting the Cloud. Design decisions for cloud computing service models (SaaS, PaaS and IaaS). John Wiley & Sons Inc

Järvinen, P. Järvinen, A. 2011. Tutkimustyön metodeista. Tampereen yliopistopaino

Miles, M.B. & Huberman, A.M. 1994. Qualitative data analysis. Thousand Oaks: Sage. Vol.14.

Nunamaker, J. Chen, M. Purdin, T. 1991. Systems Development in Information Systems Research. Journal of Management Information Systems

Sitaram, D. Manjunath, G. 2012. Moving to the cloud. Developing Apps in the New World of Cloud Computing. Syngress publications

Wiggins, A. 2012. The Twelve-Factor App, eBook

Yin, R. 2009. Case Study Research. Design and Methods. SAGE Publications Inc.

Artikkelit:

Abraham, S. Thomas, M. Thomas, J. 2005. Enhancing Web Services Availability. IEEE Computer Society

Chen, L. 2015. Determinants of Software-as-a-Service Adoption and Intention to Use for Enterprise Applications. I-Shou University

Churakova, I. Mikhramova, R. 2010. Software as a Service: Study and Analysis of SaaS Business Model and Innovation Ecosystems. University of Gent.

Davis, F. Bagozzi, P. Warshaw, P. 1989. User Acceptance of computer technology: A comparison of two theoretical models. Management Science

Eisenhardt, K. 1989. Building theories from case study research. Academy of Management Review.

Hevner, A. Chatterjee, S. 2010. Design Research in Information Systems. Springer Science+Business Media

Maged, M. Moreira, J. Shiloach, D. Wisniewski, R. 2007. Scale-up x Scale-out: A Case Study using Nutch/Lucene. IBM Thomas J. Watson Research Center. IEEE.

Tang, C. Liu, J. 2015. Selecting a trusted cloud service provider for your SaaS program. Elsevier Ltd

Yang, Z. Sun, J. Zhang, Y. Wang, Y. 2014. Understanding SaaS adoption from the perspective of organizational users: A tripod readiness model. Elsevier Ltd

Sähköiset lähteet:

Dhandala, N. 2015. 3 signs that you might need to move to NoSQL. Viitattu 11.3.2016.
<https://blog.cloudboost.io/3-signs-that-you-might-need-to-move-to-nosql/>

Gartner. 2007. Gartner says Worldwide Customer Relationship Management Software Market Will Grow 14 Percent in 2007. Viitattu 9.3.2016.
<http://www.gartner.com/newsroom/id/519316>

Gartner. 2010. Gartner Says Modernization and Digital Transformation Projects Are Behind Growth in Enterprise Application Software Market. Viitattu 9.3.2016.
<http://www.gartner.com/newsroom/id/3119717>

Gartner IT Glossary. Digitalization. Viitattu 17.5.2016. <http://www.gartner.com/it-glossary/digitalization>

IDC. 2015. Worldwide SaaS and Cloud Software 2015 – 2019 Forecast and 2014 Vendor Shares. Viitattu 9.3.2016. <https://www.idc.com/getdoc.jsp?containerId=257397>

Jordan, J. Hariharan, A. Chen, F. Kasireddy, P. 16 Startup Metrics. Viitattu 4.2.2016.
<http://a16z.com/2015/08/21/16-metrics/>

Lofgren, L. 2012. The 5 “Must Have” Metrics for Your SaaS Business. Viitattu 4.2.2016.
<https://blog.kissmetrics.com/5-metrics-for-saas/>

Mell, P. Grance, T. 2011. The NIST Definition of Cloud Computing. Viitattu 9.3.2016.
<http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>

Nadella, S. 2015. Convergence 2015. Microsoft News Center. Viitattu 22.3.2016.
<https://news.microsoft.com/speeches/satya-nadella-convergence-2015/>

Ohjelmistoyrittäjät. 2015 a. Media ja materiaalit. Viitattu 9.3.2016.
<http://www.ohjelmistoyrittajat.fi/fi/materiaalit>

Ohjelmistoyrittäjät. 2015 b. Growth with Self-Service SaaS-Marketplace Accelerator Program. Viitattu 9.3.2016. <http://www.ohjelmistoyrittajat.fi/fi/grow-self-service-saas-marketplace-accelerator-program>

Remde, K. 2011. SaaS, PaaS, and IaaS. Oh my! (“Cloudy April” – Part 3). Viitattu 9.3.2016.
<https://blogs.technet.microsoft.com/kevinremde/2011/04/03/saas-paas-and-iaas-oh-my-cloudy-april-part-3/>

Rodden, K. 2015. How to choose the right UX metrics for your product. Viitattu 4.2.2016.
<https://library.gv.com/how-to-choose-the-right-ux-metrics-for-your-product-5f46359ab5be#.1fm8pobhv>

Skok, D. SaaS Metrics 2.0 – A Guide to Measuring and Improving what Matters. Viitattu 4.2.2016. <http://www.forentrepreneurs.com/saas-metrics-2/>

Software & Information Industry Association. 2001. Software as a Service : Strategic Background. Viitattu 9.3.2016. <https://www.edocr.com/web-api/shares/83ev4lkw/download>

Statista. 2016. Global public SaaS market size 2008-2020. Viitattu 9.6.2016.
<http://www.statista.com/statistics/510333/worldwide-public-cloud-software-as-a-service/>

Työ- ja Elinkeinoministeriö. 2015. Raportti: Digitalisaatio muuttaa palvelusektoria voimakkaasti. Viitattu 17.5.2016.
https://www.tem.fi/ajankohtaista/tiedotteet/tiedotearkisto/vuosi_2015?119950_m=117806

Valtiovarainministeriö. 2015. Digitalisaatio. Viitattu 22.3.2016. <http://vm.fi/digitalisaatio>

Julkaisemattomat lähteet:

Chartmogul, 2015. The Ultimate SaaS Metrics Cheat Sheet. Version 1.2. Viitattu 3.3.2016. Dokumentti saatu sähköpostitse.

OnApp. 2016. Compete in the Cloud: Four Strategies for 2016. Viitattu 2.2.2016. Dokumentti saatu sähköpostitse.

Sihvonen, K. 2016. DevOps - Agile Efficiency and Quality. Esitys 14.3.2016.

Kuviot

Kuvio 1: A Multimethodological Approach to IS Research (Nunamaker 1991, 94).....	8
Kuvio 2: Pilvipalvelun vastuunjako (Remde 2011)	10
Kuvio 3: Global public SaaS market size 2008 – 2020 (Statista 2016)	12
Kuvio 4: Työn rakenne mukaillen Yin (1994) tapaustutkimusta (vrt. Yin 2009, 57)	15
Kuvio 5: Kuvailevan teorian rakentamisen vaiheet (Hevner 2010, 34)	16
Kuvio 6: Doing Case Study Research (Yin 2009, 1)	18
Kuvio 7: Technology Acceptance Model (Davis ym. 1989, 4).....	21
Kuvio 8: Software-as-a-Service Adoption and Intention to Use (Chen 2015, 141)	21
Kuvio 9: Tripod model of SaaS Readiness (Yang ym. 2015, 256).....	22
Kuvio 10: Scale-Up or Scale-Out (Dhandala 2015)	24
Kuvio 11: Selecting a trusted cloud service provider for your SaaS (Tang ym. 2015, 71) ..	27
Kuvio 12: DevOps Prosessi (Sihvonen 2016)	28
Kuvio 13: Online usability studies (Albert ym. 2010, 5).....	30
Kuvio 14: HEART - Goals-Signs-Metrics (Rodden 2015)	31
Kuvio 15: Two key guidelines for SaaS startups (Skok).....	33
Kuvio 16: SaaS business marketing funnel (Lofgren 2012)	33
Kuvio 17: Ohjelmistoyrityksen SaaS-valmius.....	37

Taulukot

Taulukko 1: Kriittiset menestystekijät ohjelmiston tarjoamiseen palveluna.....	34
--	----

Liitteet

Liite 1: Research Attributes	46
------------------------------------	----

Liite 1: Research Attributes

The research attributes are established for description of the level of methodological rigor in information systems research. While the level of methodological rigor has experienced different progress with respect to some specific attributes, the overall assessed rigor is currently somewhat as indeterminate and there are still significant extents for improvement. One of the keys is to include sound documentation particularly regarding issues related to the data collection and analysis processes. The followed list of attributes is based on (Davison, Martinsons, & Kock, 2004; Dubé & Paré, 2003; Locke, Spirduso, & Silverman, 2007; Miles & Huberman, 1994).

Title of study	Success factors for Providers of Software as a Service (SaaS).
Research questions	How to understand and utilize success factors for providing software as a service.
Research agreement	The researcher is permitted to use the collected research data for this thesis.
Unit of analysis	Success factor for such as: service, suitability, scalability, measurability and integration.
Importance of study	Digital transformation: software providers are moving towards providing Software-as-a-Service.
Methodological focus	Interviews workshops (n=14); literature (n=24); notes (n=16); and process flows (n=3).
Form of analysis	Mainly a qualitative analysis, saturation in interviews and data triangulation.
Research Approach	Inductive research for understanding success factors.
Specification of constructs	SaaS (Software-as-a-Service), software infrastructure, performance measurement, automated delivery, Software-as-a-Product, Information systems development (n=10).
Thesis approach	The production of software as a service.
Theoretical literature	(n=24).
First research target	Readiness for software provider to provide software as a service.
Outcome comparison	Churakova & Mikhramova 2010. Software as a Service: Study and Analysis of SaaS Business Model and Innovation Ecosystems. University of Gent.
Research design	Case Study Research Analysis (methodological literature n=12).
Data collection	The data collection of this study was cumulative, and it was systematically used for a qualitative analysis between January 2013 and March 2016. Collected data is from interviews and workshops, literature and

	discussions at work, discussions with interests groups, notes and links. Collected data includes definitions, process flows, requirements, industry best practice standards, theory, customer need and end user experience changes.
Logic of evidence	Data replication for conclusion.
Data analysis literature	Benbasat et al. 2002; Campbell and Fiske 1959; Corbin and Strauss 2013; Denzin and Lincoln 1994; Eisenhardt 1989; Gerring 2007; Miles et al. 2014; Miles & Huberman 1994; Gray 2009; Robson 2011; Silverman 2009; Walsham 2006; Yin 2014; Wynn & Williams 2012. (Literature references for analysis n=10).
Questionnaire	Yes. Mainly in workshops where has been 4-7 participants. Also one-to-one meetings in where (n=6) saturation achieved.
Questionnaire (n)	Yes. (Questions n=8).
Coding	Each interview was first transcribed and then similar or identical answers were marked with certain signs (saturation) (Corbin & Strauss, 2008).
Notes	Researcher used notes to clarify answers and response (notes n=21).
Main results	Description of the Software-as-a-Service Critical success factors and a reference frame for a software company to measure their readiness and furthered steps for providing software as a service.
Main implication	The outcomes of this study help software companies on understanding the critical success factors and to recognize the challenges and needed development to achieve the capability to provide Software-as-a-Service.
Role description	Researcher as outsider (objective) and interviewees and experts as insiders (subjective).
Research associations	Association for Information Systems (AIS); Association for Computing Machinery (ACM); and Institute of Electrical and Electronics Engineers (IEEE).